WANDER AZIMUTH IMPLEMENTATION ALGORITHM FOR A STRAPDOWN INERTIAL SYSTEM

Peter S. Maybeck

Air Force Flight Dynamics Laboratory
Wright-Patterson Air Force Base, Ohio

October 1973

AFFDL-TR-73-80

WANDER AZIMUTH IMPLEMENTATION ALGORITHM FOR A STRAPDOWN INERTIAL SYSTEM

Peter S. Maybeck

UNCLASSIFIED

Security Classification

DOCUMENT CONTROL DATA - R & D AD 784752

(Security classification of title, body of abstract and indexing annotation must be entered when the overall report is classified)

1. ORIGINATING ACTIVITY (Corporate author)

Air Force Flight Dynamics Laboratory
Air Force Systems Command
Wright-Patterson Air Force Base, Ohio 45433

2a. REPORT SECURITY CLASSIFICATION: Unclassified

2b. GROUP

3. REPORT TITLE

WANDER AZIMUTH IMPLEMENTATION ALGORITHM FOR A STRAPDOWN INERTIAL SYSTEM

4. DESCRIPTIVE NOTES (Type of report and inclusive dates)

November 1972 - March 1973

5. AUTHOR(S) (First name, middle initial, last name)

Peter S. Maybeck

6. REPORT DATE	7a. TOTAL NO. OF PAGES	7b. NO. OF REFS
October 1973	78	11

8a. CONTRACT OR GRANT NO.	9a. ORIGINATOR'S REPORT NUMBER(S)
b. PROJECT NO.	AFFDL-TR-73-80
c.	9b. OTHER REPORT NO(S) (Any other numbers that may be assigned this report)
d.	

10. DISTRIBUTION STATEMENT

Approved for public release; distribution unlimited.

11. SUPPLEMENTARY NOTES	12. SPONSORING MILITARY ACTIVITY
	Air Force Flight Dynamics Laboratory Air Force Systems Command Wright-Patterson Air Force Base, Ohio 45433

13. ABSTRACT

This report develops the algorithm for deriving attitude, heading, and navigation information from a strapdown inertial system. Beginning with the fundamental physical relationships, it develops all required equations and progresses to the onboard implementation of the algorithm.

Significant features of the algorithm include: (1) Computations performed in the wander azimuth coordinate frame to provide a system capable of operating in the polar regions; (2) Separation into four loops of different iteration rates. This maintains rapid, accurate updating of the direction cosine matrix involving vehicle attitude, while processing other information and extracting display data at appropriately slower rates; (3) Fourth order Runge-Kutta integration of quaternions, using second order rate extraction, to update the attitude direction cosine matrix; (4) Specification of the computations that require double precision for adequate performance; (5) Third order damping of the vertical channel by means of barometric altimeter data.

The applicability of this algorithm to a range of vehicle and mission environments is indicated, the required adaptations being easily performed for each particular implementation.

DD FORM 1 NOV 1473

UNCLASSIFIED

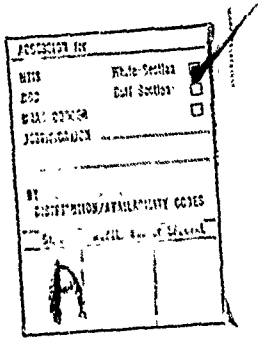
Security Classification

14. KEY WORDS	LINK A		LINK B		LINK C	
	ROLE	WT	ROLE	WT	ROLE	WT
Strapdown Inertial System						
Quaternions						
Navigation						
Attitude Algorithm						
Wander Azimuth						
Inertial System						
Gimballess (No - Gimal)						

*U.S.Government Printing Office: 1974 — 758-435/654

NOTICE

When Government drawings, specifications, or other data are used for any purpose other than in connection with a definitely related Government procurement operation, the United States Government thereby incurs no responsibility nor any obligation whatsoever; and the fact that the government may have formulated, furnished, or in any way supplied the said drawings, specifications, or other data, is not to be regarded by implication or otherwise as in any manner licensing the holder or any other person or corporation, or conveying any rights or permission to manufacture, use, or sell any patented invention that may in any way be related thereto.



Copies of this report should not be returned unless return is required by security considerations, contractual obligations, or notice on a specific document.

FOREWORD

This report was prepared by Dr. Peter S. Maybeck of the Control Elements Branch, Flight Control Division, Air Force Flight Dynamics Laboratory, Wright-Patterson Air Force Base, Ohio.

The work was initiated under Project Work Unit Number 19870248. This report covers work performed during the period from November 1972 through March 1973. The manuscript was released by the author in May 1973 for publication as a technical report.

This technical report has been reviewed and is approved.

GEORGE H. PURCELL
Chief
Control Systems Development Branch
Flight Control Division
AF Flight Dynamics Laboratory

ABSTRACT

This report develops the algorithm for deriving attitude, heading, and navigation information from a strapdown inertial system. Beginning with the fundamental physical relationships, it develops all required equations and progresses to the onboard implementation of the algorithm.

Significant features of the algorithm include:

(1) Computations performed in the wander azimuth coordinate frame to provide a system capable of operating in the polar regions;

(2) Separation into four loops of different iteration rates. This maintains rapid, accurate updating of the direction cosine matrix involving vehicle attitude, while processing other information and extracting display data at appropriately slower rates;

(3) Fourth order Runge-Kutta integration of quaternions, using second order rate extraction, to update the attitude direction cosine matrix;

(4) Specification of the computations that require double precision for adequate performance;

(5) Third order damping of the vertical channel by means of barometric altimeter data.

The applicability of this algorithm to a range of vehicle and mission environments is indicated, the required adaptations being easily performed for each particular implementation.

TABLE OF CONTENTS

SECTION		PAGE
I	INTRODUCTION	1
II	FUNDAMENTALS	3
	1. Notation	3
	2. Coordinate System Relationships: Direction Cosines and Euler Angles	7
	a. Wander Azimuth and Earth Frames	8
	b. Wander Azimuth and Body Frames	12
	3. Direction Cosine Rates	14
	4. Quaternion Updating of Direction Cosine Matrices	17
III	DEVELOPMENT OF ALGORITHM	22
	1. Conceptual Description of System	22
	2. Velocity Equations	25
	3. Angular Rate Relations	28
	4. Earth Rate Components	31
	5. Gravity Computation	31
	6. Vertical Channel Damping	32
	7. Information Extraction	35
IV	ONBOARD IMPLEMENTATION	36
	1. Integration Techniques	36
	2. Rate Extraction	38
	3. Processing Rates	39
	4. Final Algorithm	40
	a. The Partition Iterated Every T/2 Seconds	40
	b. The Partition Iterated Every T Seconds	42
	c. The Partition Iterated Every NT Seconds	49
	d. Data Extraction	54
V	USE OF ALGORITHM	58
	REFERENCES	59

LIST OF ILLUSTRATIONS

FIGURE		PAGE
1.	System Diagram for Direction Cosine Matrix	6
2.	Wander Azimuth ("Platform") and Earth Coordinate Frames	9
3.	Wander Azimuth ("Platform") and Body Coordinate Frames	13
4.	Basic System Structure	23
5.	Third Order Damped Vertical Channel	33
6.	System Block Diagram	41
7.	The T/2 Block	43
8.	Rate Extraction	44
9.	Calculation of $\underline{\omega}^b_{ip}$ and $\underline{\omega}^b_{pb}$	46
10.	$\underline{C}^p_b$ Update	47
11.	Computation of $\Delta \underline{v}^p$	50
12.	The $\underline{\ell}$ Block	51
13.	ω and $\underline{C}^p_e$ Blocks	53
14.	Calculation of $\underline{\omega}^p_{ie}$, $\underline{\omega}^p_{ip}$ and $\underline{g}$	55
15.	2NT Block Associated with $\underline{C}^p_e$	56
16.	2NT Block Associated with $\underline{C}^p_b$	57

SECTION I

INTRODUCTION

In a gimballed inertial system, the gyros and accelerometers are located on a platform whose orientation with respect to the vehicle is determined by a set of gimbals. The gyro outputs are used to drive gimbal motors in such a way as to maintain the platform in alignment with some desired coordinate frame, regardless of the vehicle orientation in space. Thus, the accelerometer outputs are coordinatized in that desired frame, and navigation information can easily be generated by performing computations in the instrumented coordinate system. Furthermore, the vehicle orientation with respect to that coordinate system can be determined by observing the angles formed between the various gimbals.

On the other hand, the gyros and accelerometers of a strapdown system are attached to the vehicle body, and therefore provide signals proportioned to specific force and angular rate between the inertial and body frames, coordinatized in the body frame. The inertial measurement unit itself is significantly simpler than a gimballed system, allowing easier maintenance and part replacement, while not being limited in performance by the imperfect response of the gimballing mechanization. Furthermore, instrument redundancy is more readily accommodated in a strapdown configuration. However, the simplicity of the strapdown system is gained at the expense of subjecting the gyros and accelerometers to a more severe environment, that of the vehicle itself. Furthermore, the computational load is greater, since the onboard computer must maintain an analytical representation of the desired coordinate frame, in addition to the calculations it would perform with a gimballed system.

This report develops the algorithm for deriving attitude, heading, and navigation information from a strapdown system. It will be implemented in an onboard computer during a strapdown development and evaluation project sponsored jointly by the Air Force Flight Dynamics Laboratory and the Army Electronics Command.

Significant features of the algorithm include:

(1) Computations performed in the wander azimuth coordinate frame to provide a system capable of operating in the polar regions;

(2) Separation into four loops of different iteration rates. This maintains rapid, accurate updating of the direction cosine matrix involving vehicle attitude, while processing other information and extracting display data at appropriately slower rates;

(3) Fourth order Runge-Kutta integration of quaternions, using second order rate extraction, to update the attitude direction cosine matrix;

(4) Specification of the computations that require double precision for adequate performance;

(5) Third order damping of the vertical channel by means of barometric altimeter data.

The report is arranged in the following manner. Section II introduces the appropriate notation and coordinate frames, and then it describes the relationships among coordinate frames by means of direction cosine matrices and Euler angles. Furthermore, the time propagation of the direction cosine matrix is described in terms of differential equations for the matrix entries and equivalently in terms of quaternions. Section III presents a conceptual description of the algorithm and then develops the details of its various segments. Section IV considers the aspects of onboard implementation, as integration methods, processing rates, and rate extraction from gyro output pulse counters. The final algorithm is also exhibited in the form of scalar equations. In conclusion, Section V indicates the applicability of this algorithm to a range of vehicle and mission environments.

SECTION II

FUNDAMENTALS

1. NOTATION

A brief description of the notation to be employed is presented in this section. It may appear somewhat cumbersome at first, but the clarity it lends to the technical developments warrants its use [References 2, 11].

Certain letters will be used as subscripts or superscripts to specify particular coordinate frames. These are:

i = inertial; with origin at the earth's center and nonrotating with respect to inertial space

e = earth; origin at the earth's center and nonrotating with respect to the earth; the unit vectors point toward the equator and the Greenwich meridian (x_e), equator and 90°E (y_e), and the North Pole (z_e).

n = north-east-down; centered at the vehicle CG and pointing north (x_n), east (y_n), and down (z_n).

p = "platform"; wander azimuth coordinate system centered at the vehicle CG and displaced from the n frame by an angle ($-\alpha$) about z_n, where the usual convention is used: that positive angles are defined as clockwise as one looks in the direction of the axis about which the rotation is made.

b = body; centered at the vehicle CG, with x_b along the longitudinal axis, y_b out the right side, and z_b out the underside of the vehicle.

Vectors are represented by an underscored lower case letter, and a superscript denotes the coordinate frame in which the vector is expressed mathematically. For example, $\underline{f}^b$ is the specific force vector expressed as a three-dimensional vector quantity coordinatized in the body (vehicle) frame. Angular velocity vectors will additionally have two subscripts,

denoting the two reference frames between which the angular velocity exists. The quantity $\underline{\omega}_{ie}^{b}$ is the angular velocity from the inertial frame to the earth frame, represented as a three-dimensional vector coordinatized in the body frame.

Components of vector quantities are denoted by the same lower case letter, without underscoring but with a subscript x, y, or z. Thus, f_x^b and ω_{iex}^{b} are the body frame x-components (longitudinal) of $\underline{f}$ and $\underline{\omega}_{ie}$ respectively.

The three-by-three direction cosine matrix that transforms a vector from one coordinate frame to another is denoted by an underscored capital C, with subscript denoting the original frame and superscript for the new frame. For instance, $\underline{C}_{b}^{i}$ is the matrix that transforms a vector from the b frame to the i frame, as

$$\underline{f}^{i} = \underline{C}_{b}^{i}\,\underline{f}^{b} \tag{1}$$

For convenience, the elements of this matrix are normally written without the subscript and superscript once the matrix itself is delineated. Equation 1 thus becomes

$$\begin{bmatrix} f_x^i \\ f_y^i \\ f_z^i \end{bmatrix} = \begin{bmatrix} C_{11} & C_{12} & C_{13} \\ C_{21} & C_{22} & C_{23} \\ C_{31} & C_{32} & C_{33} \end{bmatrix} \begin{bmatrix} f_x^b \\ f_y^b \\ f_z^b \end{bmatrix} \tag{2}$$

Here the subscripts (1, 2, 3) and (x, y, z) are interchangeable. Each element of $\underline{C}_{b}^{i}$ is a direction cosine between the unit vectors of the b and i frames. For instance, C_{12} is the direction cosine between the x (or 1) axis of the i frame and the y (or 2) axis of the b frame.

The inverse of a direction cosine matrix is simply its transpose, denoted with a superscript T:

$$\underline{f}^{b} = \underline{C}_{i}^{b}\,\underline{f}^{i} = (\underline{C}_{b}^{i})^{T}\underline{f}^{i} \tag{3}$$

The values of the entries of a direction cosine matrix can be specified completely by their initial conditions and knowledge of the angular rates between the two frames of interest. For instance, $\underline{C}_b^i(t)$ (or $\underline{C}_i^b(t)$) is specified by an initial condition and knowledge of either $\underline{\omega}_{ib}^b$ or $\underline{\omega}_{ib}^i$, the angular rate from the i frame to the b frame, coordinatized in either frame. (Note that $\underline{\omega}_{ib}^b = -\underline{\omega}_{bi}^b$, and similarly $\underline{\omega}_{ib}^i = -\underline{\omega}_{bi}^i$). In system diagrams, this dependence will be represented as in Figure 1. The angular velocity $\underline{\omega}_{ib}^b$ used to propagate the direction cosine matrix enters the block labelled $\underline{C}_b^i$ from above or below. Vectors that are to be transformed by the matrix enter the block from the side.

Writing out the propagation relations explicitly will employ a "cross-matrix". This matrix is defined to be the three-by-three matrix which, when postmultiplied by a vector coordinatized in the appropriate frame, yields the result of performing a cross-product on that vector in those coordinates. Let r^b be some vector in the b frame, and suppose one desires to compute $(\underline{\omega}_{ib} \times \underline{r})^b$; the matrix $\underline{W}_{ib}^{bk}$ is defined to satisfy

$$\underline{W}_{ib}^{bk}\underline{r}^b = (\underline{\omega}_{ib} \times \underline{r})^b \tag{4}$$

Thus, $\underline{W}_{ib}^{bk}$ is defined as

$$\underline{W}_{ib}^{bk} = \begin{bmatrix} 0 & -\omega_{ibz}^b & \omega_{iby}^b \\ \omega_{ibz}^b & 0 & -\omega_{ibx}^b \\ -\omega_{iby}^b & \omega_{ibx}^b & 0 \end{bmatrix} \tag{5}$$

where the entries are the components of the vector $\underline{\omega}_{ib}^b$.

Time derivatives of coordinatized vector quantities will be denoted by a dot above the quantity or the operator p preceding the quantity:

$$\frac{d}{dt}\underline{v}^b = \dot{\underline{v}}^b = p\underline{v}^b \tag{6}$$

With physical vectors (for instance $\underline{x}$, as opposed to $\underline{x}^b$ which is the set of three scalars that defines $\underline{x}$ in terms of body coordinates), the

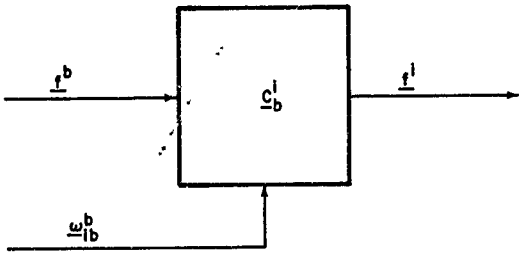


Figure 1. System Diagram for Direction Cosine Matrix

operator p will bear a subscript to depict the coordinate frame used to observe the rate of change of the vector in question. With this notation, the Theorem of Coriolis becomes, for any vector $\underline{v}$,

$$p_i \underline{v} = p_b \underline{v} + \underline{\omega}_{ib} \times \underline{v} \qquad (7)$$

2. COORDINATE SYSTEM RELATIONSHIPS: DIRECTION COSINES AND EULER ANGLES

The coordinate systems of particular interest for this report are the body (b), wander azimuth or "platform" (p), and earth (e) frames. First of all, the strapdown instruments provide information coordinatized in the body frame, while algorithm computations will be conducted in the wander azimuth frame. Further, the relationship between these two frames will specify the aircraft attitude. Finally, the relationship between the wander azimuth frame and the earth frame will determine the location of the vehicle.

There are two familiar means of describing rotations of one coordinate frame relative to another. Euler angles are a set of three rotations taken in a specified order to generate the desired orientation. The same orientation can be uniquely described by means of the nine direction cosines that exist between the unit vectors of the two frames. Whereas there is no redundancy in the Euler angle description, there are six constraints on the direction cosines (the sums of the squares of elements in a single row or column equal unity). Despite this redundancy, the direction cosine technique is preferable because the Euler angle description has an indeterminate orientation (analytical "gimbal lock") and a surrounding region of poor resolution.

The final algorithm in this report will employ two direction cosine matrices, one propagated by conventional means and the other evaluated as a function of an appropriate quaternion. Essentially, a quaternion is a means of describing angular orientation with four parameters, the minimum redundancy that removes the indeterminate points of three-parameter descriptions. Although the loop computations use direction cosines, the

data extracted for display or autopilot inputs is most conveniently expressed in the form of Euler angles. Thus, the subsequent sections will delineate the calculation of Euler angles from the corresponding direction cosine matrices.

a. Wander Azimuth and Earth Frames

Figure 2 portrays the relationship between the earth and wander azimuth frames. When the three angles, λ, L and α, are zero, the wander azimuth frame is oriented so that x_p is aligned with z_e, y_p with y_e, and z_p with $-x_e$. A positive rotation of λ, the longitude, is taken about the x_p axis. Subsequently, a negative rotation about the displaced y_p axis, of magnitude L (latitude), is made. Finally, a negative rotation about the displaced z_p axis is made, of magnitude α (wander azimuth angle). Use of a negative rotation was chosen arbitrarily to conform to the usual definition of α as a counterclockwise rotation as seen from above.

From the figure it can be seen that geographic latitude, rather than geocentric, will be used to describe position on the earth. Thus z_p is normal to the elliptical surface of the earth rather than in the direction of the earth center. Throughout the development of this report, the ellipticity of the earth will be accounted for, but higher order effects and local geoid perturbations will be neglected.

The direction cosine matrix $\underline{C}_e^p$ can be developed as the product of individual rotation matrices. First the p frame is rotated to align the axes as $x_p = z_e$, $y_p = y_e$, and $z_p = -x_e$. Then rotations of λ about x_p, $-L$ about y_p, and $-\alpha$ about z_p are performed in that order. Thus,

$$\underline{C}_e^p = \begin{bmatrix} \cos\alpha & -\sin\alpha & 0 \\ \sin\alpha & \cos\alpha & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \cos L & 0 & \sin L \\ 0 & 1 & 0 \\ -\sin L & 0 & \cos L \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\lambda & \sin\lambda \\ 0 & -\sin\lambda & \cos\lambda \end{bmatrix} \begin{bmatrix} 0 & 0 & 1 \\ 0 & 1 & 0 \\ -1 & 0 & 0 \end{bmatrix} \tag{8}$$

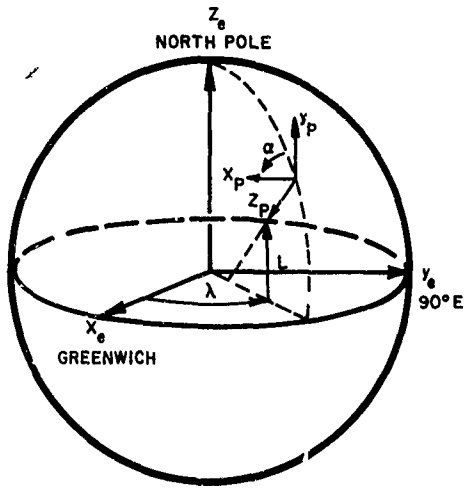


Figure 2. Wander Azimuth ("Platform") and Earth Coordinate Frames

Now if the elements of $\underline{C}_e^p$ are identified as

$$C_e^p = \begin{bmatrix} C_{11} & C_{12} & C_{13} \\ C_{21} & C_{22} & C_{23} \\ C_{31} & C_{32} & C_{33} \end{bmatrix} \tag{9}$$

then the individual elements are

$$\begin{aligned}
C_{11} &= -\cos\alpha \ \sin L \cos\lambda + \sin\alpha \sin\lambda \\
C_{12} &= -\cos\alpha \ \sin L \sin\lambda - \sin\alpha \cos\lambda \\
C_{13} &= \cos\alpha \ \cos L \\
C_{21} &= -\sin\alpha \ \sin L \cos\lambda - \cos\alpha \sin\lambda \\
C_{22} &= -\sin\alpha \ \sin L \sin\lambda + \cos\alpha \cos\lambda \\
C_{23} &= \sin\alpha \ \cos L \\
C_{31} &= -\cos L \ \cos\lambda \\
C_{32} &= -\cos L \ \sin\lambda \\
C_{33} &= -\sin L
\end{aligned} \tag{10}$$

To obtain the latitude, longitude, and wander azimuth from the elements of $\underline{C}_e^p$, the following calculations are made

$$L = \sin^{-1}(-C_{33}) \tag{11}$$

$$\lambda_p = \tan^{-1}(C_{32}/C_{31}) \tag{12}$$

$$\alpha_p = \tan^{-1}(C_{23}/C_{13}) \tag{13}$$

The subscript p denotes principal values in Equations 12 and 13. Latitude is defined in the range (-90°, 90°), which coincides with the principal values of the $\sin^{-1}$ function, and thus there is no ambiguity to the solution of Equation 11. However, λ is defined over (-180°, 180°) and α over (0°, 360°), whereas the principal values of the $\tan^{-1}$ function lie within (-90°, 90°). The ambiguity can be resolved by observing the sign of the elements C_{31} and C_{13}. Since cos L is nonnegative, these elements yield the sign of -cos λ and cos α respectively. Table I demonstrates the evaluation of the true λ and α values as a function of these signs.

TABLE I

LONGITUDE AND WANDER AZIMUTH EVALUATIONS

LONGITUDE (λ)

Sign of λ_p	Sign of C_{31}	λ Evaluation	Quadrant
+	+	λ_p-180°	(-180°, -90°)
-	-	λ_p	(-90°, 0°)
+	-	λ_p	(0°, 90°)
-	+	λ_p+180°	(90°, 180°)

WANDER AZIMUTH (α)

Sign of α_p	Sign of C_{13}	α Evaluation	Quadrant
+	+	α_p	(0°, 90°)
-	-	α_p+180°	(90°, 180°)
+	-	α_p+180°	(180°, 270°)
-	+	α_p+360°	(270°, 360°)

Using conventional propagation of the direction cosine matrix, it is possible to neglect the propagation of C_{11}, C_{21}, and C_{31} if these values are not required for other computations. The elements C_{11} and C_{21} will not be used at all, and C_{31} is required only to compute the value of longitude. However, by manipulating the relations of Equation 10, it can be shown that

$$C_{31} = C_{23}C_{12} - C_{22}C_{13} \tag{14}$$

Thus, only six of the nine direction cosines will be propagated, and Equation 14 used to calculate the value of C_{31} for the longitude evaluation.

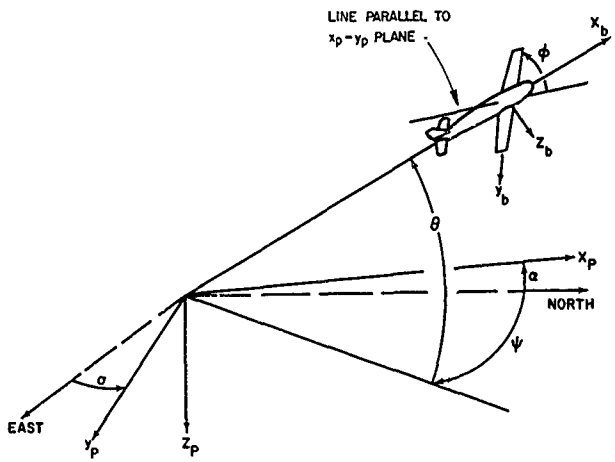
b. Wander Azimuth and Body Frames

Figure 3 presents the relationships between the wander azimuth and body frames. Rotating the wander azimuth frame through a positive rotation of α about z_p yields a north-east-down coordinate frame. From there the ordinary Euler angles of yaw (ψ), pitch (θ), and roll (ϕ) yield the aircraft attitude. Consequently, the matrix $\underline{C}_p^b$ can be composed of the product of three rotation matrices, ($\psi + \alpha$) about z_b, then θ about the displaced y_b, and ϕ about x_b:

$$\underline{C}_p^b = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\phi & \sin\phi \\ 0 & -\sin\phi & \cos\phi \end{bmatrix} \begin{bmatrix} \cos\theta & 0 & -\sin\theta \\ 0 & 1 & 0 \\ \sin\theta & 0 & \cos\theta \end{bmatrix} \begin{bmatrix} \cos(\psi+\alpha) & \sin(\psi+\alpha) & 0 \\ -\sin(\psi+\alpha) & \cos(\psi+\alpha) & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{15}$$

The inverse transformation will be of direct interest, so define

$$\underline{C}_b^p = \begin{bmatrix} T_{11} & T_{12} & T_{13} \\ T_{21} & T_{22} & T_{23} \\ T_{31} & T_{32} & T_{33} \end{bmatrix} = \underline{C}_p^{bT} \tag{16}$$



NOTE: ORIGIN OF p FRAME DISPLACED FROM THAT OF b FRAME ONLY FOR CLARITY OF DIAGRAM; THEY ARE ACTUALLY COINCIDENT AT VEHICLE C3.

Figure 3. Wander Azimuth ("Platform") and Body Coordinate Frames

Using this definition, the individual elements become

$$
\begin{aligned}
T_{11} &= \cos(\psi+\alpha)\cos\theta \\
T_{12} &= \cos(\psi+\alpha)\sin\theta\ \sin\phi - \sin(\psi+\alpha)\cos\phi \\
T_{13} &= \cos(\psi+\alpha)\sin\theta\ \cos\phi + \sin(\psi+\alpha)\ \sin\phi \\
T_{21} &= \sin(\psi+\alpha)\cos\theta \\
T_{22} &= \sin(\psi+\alpha)\sin\theta\ \sin\phi + \cos(\psi+\alpha)\cos\phi \\
T_{23} &= \sin(\psi+\alpha)\sin\theta\ \cos\phi - \cos(\psi+\alpha)\cos\phi \\
T_{31} &= -\sin\theta \\
T_{32} &= \cos\theta\ \sin\phi \\
T_{33} &= \cos\theta\ \cos\phi
\end{aligned}
\tag{17}
$$

The values of $(\psi + \alpha)$, θ, and ϕ are obtained from the elements of $\underline{C}_b^p$ in a manner similar to the data extraction of the previous section:

$$\theta = \sin^{-1}(-T_{31}) \tag{18}$$

$$\phi_p = \tan^{-1}(T_{32}/T_{33}) \tag{19}$$

$$(\Psi+\alpha)_p = \tan^{-1}(T_{21}/T_{11}) \tag{20}$$

Pitch is defined over the range (-90°, 90°) and no ambiguities exist. However, roll is defined over (-180°, 180°) and $(\psi + \alpha)$ over (0°, 360°). Cos θ is nonnegative, so the ambiguity can be resolved with the aid of the sign of T_{33} and T_{11}, as in Table II.

Once the value of $(\psi + \alpha)$ is obtained, it can be combined with the value of α derived in the previous section to obtain yaw:

$$\Psi = (\Psi+\alpha) - \alpha \tag{21}$$

If the result is a negative number, 360° can be added to it to yield a value of ψ in the range of (0°, 360°).

3. DIRECTION COSINE RATES

In order to propagate the values of the direction cosine matrix elements in the conventional manner, an expression for their rate of

TABLE II

ROLL AND YAW-PLUS-WANDER AZIMUTH EVALUATIONS

ROLL (ϕ)

Sign of ϕ_p	Sign of T_{33}	ϕ Evaluation	Quadrant
+	-	ϕ_p-180°	(-180°, -90°)
-	+	ϕ_p	(-90°, 0°)
+	+	ϕ_p	(0°, 90°)
-	-	ϕ_p + 180°	(90°, 180°)

YAW-PLUS-WANDER AZIMUTH ($\psi + \alpha$)

Sign of $(\Psi + \alpha)_p$	Sign of T_{11}	$(\psi + \alpha)_p$ Evaluation	Quadrant
+	+	$(\psi + \alpha)_p$	(0°, 90°)
-	-	$(\psi + \alpha)_p$ + 180°	(90°, 180°)
+	-	$(\psi + \alpha)_p$ + 180°	(180°, 270°)
-	+	$(\psi + \alpha)_p$ + 360°	(270°, 360°)

change is needed. For any two frames, a and b, the Theorem of Coriolis can be written for any vector $\underline{v}$ as

$$P_a \underline{v} = P_b \underline{v} + \underline{\omega}_{ab} \times \underline{v} \tag{22}$$

Coordinatize the left side of this equation in the b frame to yield

$$\begin{aligned} (P_a \underline{v})^b &= \underline{C}_a^b \left[p\underline{v}^a \right] \\ &= \underline{C}_a^b \left[p(\underline{C}_b^a \underline{v}^b) \right] \\ &= \underline{C}_a^b \left[(p\underline{C}_b^a) \underline{v}^b + \underline{C}_b^a p\underline{v}^b \right] \\ &= \underline{C}_a^b (p\underline{C}_b^a) \underline{v}^b + p\underline{v}^b \end{aligned} \tag{23}$$

Equating this to the right side coordinatized in the b frame yields

$$p\underline{v}^b + \underline{C}_a^b \left(p\underline{C}_b^a \right) \underline{v}^b = p\underline{v}^b + \underline{W}_{ab}^{bk} \underline{v}^b \tag{24}$$

Since this is true for all vectors $\underline{v}^b$, Equation 24 implies

$$\dot{\underline{C}}_b^a = \underline{C}_b^a \underline{W}_{ab}^{bk} \tag{25}$$

where

$$\underline{W}_{ab}^{bk} = \begin{bmatrix} 0 & -\omega_z & \omega_y \\ \omega_z & 0 & -\omega_x \\ -\omega_y & \omega_x & 0 \end{bmatrix} \tag{26}$$

$$\begin{bmatrix} \omega_x \\ \omega_y \\ \omega_z \end{bmatrix} = \underline{\omega}_{ab}^b = -\underline{\omega}_{ba}^b \tag{27}$$

These are the desired relations, expressible as nine scalar equations.

If the available angular rate information were coordinatized in the a frame instead of the b frame, the desired relations become

$$\dot{\underline{C}}_b^a = \underline{W}_{ab}^{ak}\,\underline{C}_b^a \tag{28}$$

where $\underline{W}_{ab}^{ak}$ is defined as in Equation 26, except that the entries are components of $\underline{\omega}_{ab}^{a}$ ($= -\underline{\omega}_{ba}^{a}$). To show this, write Equation 25 with the roles of a and b reversed:

$$\dot{\underline{C}}_a^b = \underline{C}_a^b\,\underline{W}_{ba}^{ak}$$

Now use the facts that $\underline{C}_a^b = \underline{C}_b^{aT}$ and $\underline{\omega}_{ba}^{a} = -\underline{\omega}_{ab}^{a}$ to write

$$\left(\dot{\underline{C}}_b^{aT}\right) = \left(\dot{\underline{C}}_b^a\right)^T = \underline{C}_b^{aT}\,\underline{W}_{ba}^{ak} = -\underline{C}_b^{aT}\,\underline{W}_{ab}^{ak}$$

Transposing yields

$$\dot{\underline{C}}_b^a = -\underline{W}_{ab}^{ak\ T}\,\underline{C}_b^a$$

$$= \underline{W}_{ab}^{ak}\,\underline{C}_b^a$$

which is Equation 28.

It has been common practice to solve these equations with a first order integration technique using DDA's to achieve high iteration rates (on the order of 10,000 per second) and thereby maintain accuracy. The current algorithm is designed for whole word computers, using a slower iteration rate but a higher order integration method where necessary for precision. Because of the amount of computation involved in these higher order methods, it is more efficient to propagate four quaternion parameters instead of nine direction cosines, and then calculate the matrix entries algebraically.

4. QUATERNION UPDATING OF DIRECTION COSINE MATRICES

As mentioned previously, a rotation of one frame relative to another can be expressed uniquely in terms of four parameters, three to define

the axis of rotation and the fourth to specify the amount of rotation. A considerable amount of quaternion algebra can be developed to provide these expressions [References 5, 7, 9], but it is more convenient for the purpose of this algorithm to define the direction cosine matrix in terms of the propagated quaternions [References 6, 7, 10]. Coordinate transformations are then performed in the conventional manner.

Let A, B, C, and D be the four quaternion parameters. Conceptually, A, B, and C will define a vector in space and D will be the amount of rotation about that vector. With appropriate definitions of these four parameters, a general direction cosine matrix, $\underline{C}\,{}^{a}_{b}$ as in the previous section, can be evaluated as

$$\underline{C}\,{}^{a}_{b} = \begin{bmatrix} A^2 - B^2 - C^2 + D^2 & 2(AB + CD) & 2(AC - BD) \\ 2(AB - CD) & -A^2 + B^2 - C^2 + D^2 & 2(BC + AD) \\ 2(AC + BD) & 2(BC - AD) & -A^2 - B^2 + C^2 + D^2 \end{bmatrix} \tag{29}$$

Note that changing the sign of D yields $\underline{C}\,{}^{aT}_{b} = \underline{C}\,{}^{b}_{a}$. This is expected since the negative of the rotation about the unit vector used to obtain frame a should yield the original frame b again. Furthermore, changing the sign of all four quaternion parameters leaves $\underline{C}\,{}^{a}_{b}$ unchanged. This, too, is reasonable: a positive rotation about a certain vector equals the negative rotation of same magnitude about the vector in the opposite direction. This fact will be utilized in the initialization of the quaternions.

Now a rate equation analogous to Equation 25 will be developed. Given the definition Equation 29 and the angular rate information $\underline{\omega}\,{}^{b}_{ab} = -\underline{\omega}\,{}^{b}_{ba}$, the appropriate rate equation can be derived as

$$\begin{bmatrix} \dot{A} \\ \dot{B} \\ \dot{C} \\ \dot{D} \end{bmatrix} = \frac{1}{2} \begin{bmatrix} 0 & \omega_z & -\omega_y & -\omega_x \\ -\omega_z & 0 & \omega_x & -\omega_y \\ \omega_y & -\omega_x & 0 & -\omega_z \\ \omega_x & \omega_y & \omega_z & 0 \end{bmatrix} \begin{bmatrix} A \\ B \\ C \\ D \end{bmatrix} \tag{30}$$

where

$$\begin{bmatrix} \omega_x \\ \omega_y \\ \omega_z \end{bmatrix} = \underline{\omega}^b_{ab} = -\underline{\omega}^b_{ba} \tag{31}$$

If instead the angular rate data were available in a-frame coordinates, the relation is

$$\begin{bmatrix} \dot{A} \\ \dot{B} \\ \dot{C} \\ \dot{D} \end{bmatrix} = \frac{1}{2} \begin{bmatrix} 0 & -\omega_z & \omega_y & -\omega_x \\ \omega_z & 0 & -\omega_x & -\omega_y \\ -\omega_y & \omega_x & 0 & -\omega_z \\ \omega_x & \omega_y & \omega_z & 0 \end{bmatrix} \begin{bmatrix} A \\ B \\ C \\ D \end{bmatrix} \tag{32}$$

where

$$\begin{bmatrix} \omega_x \\ \omega_y \\ \omega_z \end{bmatrix} = \underline{\omega}^a_{ab} = -\underline{\omega}^a_{ba} \tag{33}$$

Note that Equation 30 or 32 would also be satisfied if the sign of all four parameters were reversed; again this is useful for initialization considerations.

Since there is a single redundancy in a four parameter description of coordinate rotations (as opposed to six for the direction cosines), the quaternion parameters satisfy a single constraint equation:

$$A^2 + B^2 + C^2 + D^2 = 1 \tag{34}$$

Computationally, $(A^2 + B^2 + C^2 + D^2)^{1/2}$ can be used as a normalizing factor for each parameter; this is analogous to the periodic orthogonalization procedure used in conjunction with direction cosine propagations.

Quaternions will be employed to update $\underline{C}\,{}^{p}_{b}$. The b frame is obtained from the p frame by a rotation of $(\Psi + \alpha)$ about z_b, θ about y_b, and finally ϕ about x_b. An appropriate definition of A, B, C, and D that yields $\underline{C}\,{}^{p}_{b}$ according to Equation 29 and also satisfies Equation 34 is:

$$A = \sin\frac{\Psi+\alpha}{2}\sin\frac{\theta}{2}\cos\frac{\phi}{2} - \cos\frac{\Psi+\alpha}{2}\cos\frac{\theta}{2}\sin\frac{\phi}{2} \quad (35a)$$

$$B = -\cos\frac{\Psi+\alpha}{2}\sin\frac{\theta}{2}\cos\frac{\phi}{2} - \sin\frac{\Psi+\alpha}{2}\cos\frac{\theta}{2}\sin\frac{\phi}{2} \quad (35b)$$

$$C = -\sin\frac{\Psi+\alpha}{2}\cos\frac{\theta}{2}\cos\frac{\phi}{2} + \cos\frac{\Psi+\alpha}{2}\sin\frac{\theta}{2}\sin\frac{\phi}{2} \quad (35c)$$

$$D = \cos\frac{\Psi+\alpha}{2}\cos\frac{\theta}{2}\cos\frac{\phi}{2} + \sin\frac{\Psi+\alpha}{2}\sin\frac{\theta}{2}\sin\frac{\phi}{2} \quad (35d)$$

Although these relations define the quaternion parameters, they are not convenient for initialization. Besides requiring sine and cosine evaluations, the triple products of those functions suffer from potential numerical inaccuracies on a short wordlength computer. Moreover, typical initialization procedures yield the direction cosines directly, with Euler angles computed from these. Thus, the initial magnitudes of A, B, C, and D can be defined more appropriately in terms of elements of $\underline{C}\,{}^{p}_{b}$:

$$|A| = \frac{1}{2}\sqrt{1 + T_{11} - T_{22} - T_{33}} \quad (36a)$$

$$|B| = \frac{1}{2}\sqrt{1 - T_{11} + T_{22} - T_{33}} \quad (36b)$$

$$|C| = \frac{1}{2}\sqrt{1 - T_{11} - T_{22} + T_{33}} \quad (36c)$$

$$|D| = \sqrt{1 - A^2 - B^2 - C^2} \quad (36d)$$

The first three result from algebraic manipulation of the diagonal terms of Equation 29, and the last is just a restatement of the constraint Equation 34. The signs of these terms need to be established. Taking the difference of off-diagonal terms in Equation 29 yields

$$4AD = T_{23} - T_{32} \tag{37a}$$

$$4BD = T_{31} - T_{13} \tag{37b}$$

$$4CD = T_{12} - T_{21} \tag{37c}$$

Thus, once the sign of D is set, these three relations provide the signs of the other three parameters. It was previously demonstrated that the choice is, in fact, arbitrary. Therefore, the signs are chosen as

$$\text{sign } D = + \tag{38a}$$

$$\text{sign } A = \text{sign } (T_{23} - T_{32}) \tag{38b}$$

$$\text{sign } B = \text{sign } (T_{31} - T_{13}) \tag{38c}$$

$$\text{sign } C = \text{sign } (T_{12} - T_{21}) \tag{38d}$$

At initialization time, Equations 36 and 38 are used to evaluate A, B, C, and D from the value of $\underline{C}\,{}^{p}_{b}$ established in alignment. In the operational mode, $\underline{\omega}\,{}^{b}_{pb}$ is available so the parameters are updated according to Equation 30, with Equation 31 defined as $\underline{\omega}\,{}^{b}_{pb}$. When $\underline{C}\,{}^{p}_{b}$ is required, it is evaluated by means of Equation 29.

SECTION III

DEVELOPMENT OF ALGORITHM

1. CONCEPTUAL DESCRIPTION OF SYSTEM

The basic computational system structure is as depicted in Figure 4. Body mounted accelerometers measure $\underline{a}^b$, the specific force exerted on the instruments by their supports (vehicle acceleration minus the acceleration of mass attraction gravity) along body coordinates. This is simply the negative of $\underline{f}^b$, conventionally defined as the specific force exerted by the instrument set on its supports. Similarly, gyros provide a measure of the angular rate between inertial space and the body frame, expressed as components along body axes.

First the specific force $\underline{a}^b$ is transformed by the direction cosine matrix $\underline{C}_b^p$ into wander azimuth coordinates:

$$\underline{a}^p = \underline{C}_b^p \, \underline{a}^b \tag{39}$$

A current value for $\underline{C}_b^p$ is maintained through propagations using $\underline{\omega}_{pb}^b$, which is generated as

$$\begin{aligned} \underline{\omega}_{pb}^b &= \underline{\omega}_{Ib}^b - \underline{\omega}_{Ip}^b \\ &= \underline{\omega}_{Ib}^b - \underline{C}_p^b \, \underline{\omega}_{Ip}^p \\ &= \omega_{Ib}^b - \underline{C}_b^{pT} \, \underline{\omega}_{Ip}^p \end{aligned} \tag{40}$$

The transformed accelerometer outputs, $\underline{a}^p$, are then processed through the block labelled "$\mathscr{L}$" in Figure 4 to obtain $\underline{v}^p$, the vehicle linear velocity expressed in local wander azimuth coordinates. Section III.2 will develop the appropriate level loop relationships. Required by these computations are $\underline{\omega}_{ep}^p$ and $\underline{\omega}_{ie}^p$, discussed in Sections III.3 and III.4 respectively. The vertical channel additionally requires computation of g^p and damping provided by a barometric altimeter, considered in Sections III.5 and III.6.

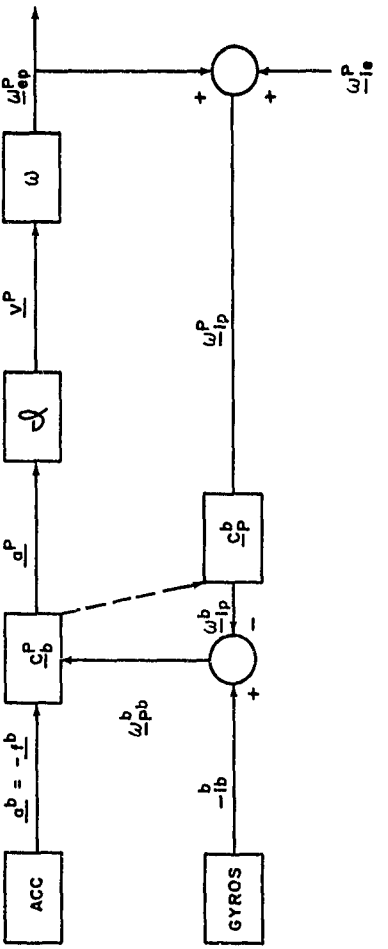


Figure 4. Basic System Structure

Once $\underline{v}^p$ is formed, it can be used to derive the value of the angular rate $\underline{\omega}^p_{ep}$; the block labelled ω is delineated in Section III.3. For the wander azimuth system to be employed (sometimes termed free azimuth), ω^p_{epz} is identically zero.

The angular rate between the inertial and wander azimuth frames, $\underline{\omega}^p_{ip}$, is then formed as

$$\underline{\omega}^p_{ip} = \underline{\omega}^p_{ep} + \underline{\omega}^p_{ie} \tag{41}$$

where $\underline{\omega}^p_{ie}$ is calculated as in Section III.4. This is the quantity used in Equation 40 and the loop is complete.

A number of these computations are functionally dependent upon latitude and longitude. A direct evaluation of these is possible by solving for α according to the relation

$$\dot{\alpha} = -\dot{\lambda} \sin L \tag{42}$$

to form $\underline{C}^n_p$ as

$$\underline{C}^n_p = \begin{bmatrix} \cos\alpha & \sin\alpha & 0 \\ -\sin\alpha & \cos\alpha & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{43}$$

which in turn is used to transform $\underline{\omega}^p_{ep}$ to $\underline{\omega}^n_{ep}$, with x and y components

$$\omega^n_{epx} = \dot{\lambda} \cos L \tag{44a}$$

$$\omega^n_{epy} = -\dot{L} \tag{44b}$$

which can be integrated to evaluate L and λ. However, the functional relationships required are easily generated from the elements of $\underline{C}^p_e$.

Since this involves a linear propagation equation without the inherent problems of direct integration of α, L and λ, it is more efficient to use $\underline{\omega}_{ep}^{p}$ to update $\underline{C}_{e}^{p}$. Then L, λ, α, and the desired functions are computed from the direction cosines.

In a similar manner, $(\psi + \alpha)$, θ, and ϕ are extracted from $\underline{C}_{b}^{p}$, while these values are not used explicitly in the computational loop. Section III.7 considers the extraction of data for presentation to either the pilot or autopilot.

2. VELOCITY EQUATIONS

The equation that relates $\underline{v}^{p}$ to $\underline{a}^{p}$ is

$$\dot{\underline{v}}^{p} = \underline{a}^{p} + \underline{g}^{p} - \left(2\underline{\omega}_{ie}^{p} + \underline{\omega}_{e}^{p}\right) \times \underline{v}^{p} \qquad (45)$$

This relation is derived in the following manner (References 4, 8). From Newtonian mechanics, the second derivative of position, as seen from the inertial frame i, satisfies the equation

$$\rho_{i}^{2}\, \underline{r} = \underline{a} + \underline{g}_{m} \qquad (46)$$

where $\underline{r}$ is the vehicle position vector, $\underline{a}$ is the output of the accelerometers, and $\underline{g}_{m}$ is the mass attraction vector. If $\underline{g}$ is the gravity vector composed of both mass attraction and centripetal components,

$$\underline{g} = \underline{g}_{m} - \underline{\omega}_{ie} \times \left(\underline{\omega}_{ie} \times \underline{r}\right) \qquad (47)$$

then Equation (46) can be written as

$$\rho_{i}^{2}\, \underline{r} = \underline{a} + \underline{g} + \underline{\omega}_{ie} \times \left(\underline{\omega}_{ie} \times \underline{r}\right) \qquad (48)$$

It is desired to write Equation 48 in terms of the rate of change of the velocity vector $\underline{v}$, as seen from the computational frame p. By

definition, the velocity $\underline{v}$ is the time rate of change of the position vector $\underline{r}$, as seen from the earth frame e:

$$\underline{v} = p_e \underline{r} \tag{49}$$

This can be related to the inertial rate of change of $\underline{r}$ by means of the Theorem of Coriolis:

$$\begin{aligned} p_i \underline{r} &= p_e \underline{r} + \underline{\omega}_{ie} \times \underline{r} \\ &= \underline{v} + \underline{\omega}_{ie} \times \underline{r} \end{aligned} \tag{50}$$

Differentiating this equation again yields

$$\begin{aligned} p_i^2 \underline{r} &= p_i \underline{v} + \underline{\omega}_{is} \times p_i \underline{r} \\ &= p_i \underline{v} + \underline{\omega}_{ie} \times \underline{v} + \underline{\omega}_{ie} \times \left(\underline{\omega}_{ie} \times \underline{r} \right) \end{aligned} \tag{51}$$

Substitute this into Equation 48 to obtain

$$p_i \underline{v} = \underline{a} + \underline{g} - \underline{\omega}_{ie} \times \underline{v} \tag{52}$$

Again using the Theorem of Coriolis, $p_i \underline{v}$ can be expressed in terms of $p_p \underline{v}$, the time derivative of $\underline{v}$ as seen in the computational frame p:

$$p_i \underline{v} = p_p \underline{v} + \underline{\omega}_{ip} \times \underline{v} \tag{53}$$

Combining Equations 52 and 53 yields

$$\begin{aligned} p_p \underline{v} &= \underline{a} + \underline{g} - \left(\underline{\omega}_{ie} + \underline{\omega}_{ip} \right) \times \underline{v} \\ &= \underline{a} + \underline{g} - \left(2\underline{\omega}_{ie} + \underline{\omega}_{ep} \right) \times \underline{v} \end{aligned} \tag{54}$$

When coordinatized in the p frame, this is the desired relation, (Equation 45).

Writing Equation 45 in scalar form yields the relations to be implemented in the algorithm:

$$\dot{v}_x^p = a_x^p + v_y^p \omega_z^p - v_z^p \omega_y^p \tag{55a}$$

$$\dot{v}_y^p = a_y^p + v_z^p \omega_x^p - v_x^p \omega_z^p \tag{55b}$$

$$\dot{v}_z^p = a_z^p + g + v_x^p \omega_y^p - v_y^p \omega_x^p \tag{55c}$$

where the quantities ω_x^p, ω_y^p, and ω_z^p are defined as

$$\omega_x^p = 2\omega_{iex}^p + \omega_{epx}^p \tag{56a}$$

$$\omega_y^p = 2\omega_{iey}^p + \omega_{epy}^p \tag{56b}$$

$$\omega_z^p = 2\omega_{iez}^p \tag{56c}$$

Note that, for the wander azimuth system being used, ω_{epz}^p is zero. Also, the approximation that $\underline{g}$ is colinear with z_p has been made:

$$\underline{g}^p \cong \begin{bmatrix} 0 \\ 0 \\ g \end{bmatrix} \tag{57}$$

This will introduce only negligible errors, on the order of resolution capabilities of present day instruments.

Equation 55c is the equation for an unaided inertial vertical channel. The computations are unstable, and so this equation will be modified in Section III.6 to provide damping by means of barometric altimeter data.

3. ANGULAR RATE RELATIONS

Let r_o be the equatorial radius of the earth and h be the vehicle altitude. If a spherical earth were assumed, the angular rate relations would simply be

$$\omega^p_{epx} = \frac{1}{r_o+h} v^p_y \tag{58a}$$

$$\omega^p_{epy} = -\frac{1}{r_o+h} v^p_x \tag{58b}$$

However, it is desired to account for the earth's ellipticity, so the relations become somewhat more complex.

Consider the north-east-down (n) frame first. With respect to this frame, the desired relations are

$$\omega^n_{enx} = \frac{1}{r_e+h} v^n_y \tag{59a}$$

$$\omega^n_{eny} = -\frac{1}{r_n+h} v^n_x \tag{59b}$$

where r_n is the radius of curvature of the reference ellipsoid in the meridian plane at a given point on the surface, and r_e is the radius of curvature in the plane orthogonal to the meridian plane at that same point. Thus, the terms $(r_n + h)$ and $(r_e + h)$ are the corresponding radii of curvature at altitude h (h defined along z_n or z_p is colinear with both r_n and r_e). The values of r_n and r_e can be shown to be [Reference 3].

$$r_n = \frac{r_o(1-\epsilon^2)}{(1-\epsilon^2\sin^2 L)^{3/2}} \tag{60a}$$

$$r_e = \frac{r_o}{(1-\epsilon^2\sin^2 L)^{1/2}} \tag{60b}$$

where ϵ is the eccentricity of the reference ellipsoid,

$$\epsilon = \frac{r_o^2 - r_p^2}{r_o^2} \tag{61}$$

r_p being defined as the polar radius.

The vehicle angular rates expressed in the n and p frames are related by

$$\omega^p_{epx} = \omega^n_{enx} \cos\alpha - \omega^n_{eny} \sin\alpha \tag{62a}$$

$$\omega^p_{epy} = \omega^n_{enx} \sin\alpha + \omega^n_{eny} \cos\alpha \tag{62b}$$

and similarly the linear velocities are related by

$$v^n_x = v^p_x \cos\alpha + v^p_y \sin\alpha \tag{63a}$$

$$v^n_y = -v^p_x \sin\alpha + v^p_y \cos\alpha \tag{63b}$$

Substituting Equation 63 into Equation 59, and then substituting the results into Equation 62, yields

$$\omega^p_{epx} = \frac{1}{R_y} v^p_y + k\, v^p_x \tag{64a}$$

$$\omega^p_{epy} = -\frac{1}{R_x} v^p_x - k\, v^p_y \tag{64b}$$

where

$$\frac{1}{R_x} = \frac{\cos^2\alpha}{r_n+h} + \frac{\sin^2\alpha}{r_e+h} \tag{65a}$$

$$\frac{1}{R_y} = \frac{\sin^2\alpha}{r_n+h} + \frac{\cos^2\alpha}{r_e+h} \tag{65b}$$

$$k = \left(\frac{1}{r_n+h} - \frac{1}{r_e+h}\right) \sin\alpha \cos\alpha \tag{65c}$$

Equations 64 and 65 provide the exact expressions for vehicle angular rates in terms of linear velocities, assuming an oblate spheroid for the earth (i.e., with elliptical meridian planes).

First order approximations to these equations can be derived by expanding $(r_n + h)^{-1}$ and $(r_e + h)^{-1}$ and retaining only first order terms to obtain [Reference 3]:

$$\frac{1}{r_n+h} \cong \frac{1}{r_0}\left(1-\frac{h}{r_0}+\epsilon^2-\frac{3}{2}\epsilon^2\sin^2 L\right) \tag{66a}$$

$$\frac{1}{r_e+h} \cong \frac{1}{r_0}\left(1-\frac{h}{r_0}-\frac{1}{2}\epsilon^2\sin^2 L\right) \tag{66b}$$

These approximations are put into Equation 65, and $1/2\ \epsilon^2$ is approximated by the ellipticity e, where

$$e = \frac{r_0 - r_p}{r_0} \tag{67}$$

yielding the first order expressions

$$\frac{1}{R_x} \cong \frac{1}{r_0}\left[1-\frac{h}{r_0}-e(1-3\cos^2 L\cos^2\alpha-\cos^2 L\sin^2\alpha)\right] \tag{68a}$$

$$\frac{1}{R_y} \cong \frac{1}{r_0}\left[1-\frac{h}{r_0}-e(1-3\cos^2 L\sin^2\alpha-\cos^2 L\cos^2\alpha)\right] \tag{68b}$$

$$k \cong \frac{2e}{r_0}\cos^2 L\sin\alpha\cos\alpha \tag{68c}$$

These evaluations can be expressed conveniently in terms of the elements of the direction cosine matrix $\underline{C}\,{}^{p}_{e}$:

$$\frac{1}{R_x} \cong \frac{1}{r_0}\left[1-\frac{h}{r_0}-e(1-3C_{13}^2-C_{23}^2)\right] \tag{69a}$$

$$\frac{1}{R_y} \cong \frac{1}{r_0}\left[1-\frac{h}{r_0}-e(1-3C_{23}^2-C_{13}^2)\right] \tag{69b}$$

$$k \cong \frac{2e}{r_0}C_{13}C_{23} \tag{69c}$$

The algorithm to be implemented computes R_x^{-1}, R_y^{-1}, and k according to Equation 69, and then calculates ω^{p}_{epx} and ω^{p}_{epy} through Equation 64.

4. EARTH RATE COMPONENTS

In the earth (e) coordinate system, the earth rotational rate vector is

$$\underline{\omega}_{ie}^{e} = \begin{bmatrix} 0 \\ 0 \\ \omega_{ie} \end{bmatrix} \tag{70}$$

Thus, in wander azimuth coordinates, this vector becomes

$$\omega_{ie}^{p} = \underline{C}_{e}^{p}\, \underline{\omega}_{ie}^{e} = \begin{bmatrix} \omega_{ie} C_{13} \\ \omega_{ie} C_{23} \\ \omega_{ie} C_{33} \end{bmatrix} \tag{71}$$

5. GRAVITY COMPUTATION

The standard approximation to the value of gravity (Reference 3) is given by:

$$g = \frac{GM}{r_0^2}\left[(1 + \tfrac{3}{2} J_2 - \tfrac{15}{8} J_4 - C) + \right.$$

$$+ (\epsilon^2 - \tfrac{1}{2} C\epsilon^2 + 15 J_2 \epsilon^2 - \tfrac{3}{4}\epsilon^4 - \tfrac{9}{2} J_2 + \tfrac{75}{4} J_4 + C)\sin^2 L$$

$$\left. -(2 + 6J_2 + C)\frac{h}{r_0}\right] \tag{72}$$

where G is the gravitational constant, M is the mass of the earth, J_2 and J_4 are coefficients of the first two harmonics of the mass attraction potential function, and C is equal to $\omega_{ie}^2 r_0^3/(GM)$. Substituting the numerical values involved and using C_{33} = -sin L, Equation 72 is put into final form as

$$g = 32.087437360 + 0.16994493815 C_{33}^2 - 0.30877321095 \times 10^{-5} h_B \ (\text{ft/sec}^2) \tag{73}$$

The subscript B on h_B (measured in feet) denotes the fact that barometric altitude is used to evaluate g in order to stabilize the vertical channel computations. This will be discussed further in the following section.

6. VERTICAL CHANNEL DAMPING

Barometric altimeter data is combined with inertial vertical channel computations to damp and bound the errors in the latter. A third order damping technique is implemented as in Figure 5. The inertial system input is $(a_z^p + \omega_y^p v_x^p - \omega_x^p v_y^p)$, where a_z^p is provided by Equation 39 and the other terms are evaluated as in Section III.2. Added to this is the computed value of g. The altitude h_B provided by the barometric altimeter is subtracted from h, the inertially indicated altitude, to generate the error signal for feedback (note that h is positive up). The vertical velocity is passed through the gain C_4 and added to h (since v_z^p is positive down) in order to compensate for the inherent lag in the barometric altimeter. Thus, typical values of C_4 would range between 0.5 and 0.8 seconds. The feedback of h through $2\omega_s^2$ compensates for changes in gravity with altitude, and it can be recognized as the cause of instability in the unaided inertial vertical channel. Although a simpler second order damping system would result from setting C_3 equal to zero, the performance advantage of the more complex system is sufficient to warrant its use.

From Figure 5, the state space model of the vertical computational loop is

$$\begin{bmatrix} \dot{h} \\ \dot{v}_z^p \\ \dot{x} \end{bmatrix} = \begin{bmatrix} -C_1 & -(1 + C_1C_4) & 0 \\ C_2 - 2\omega_s^2 & C_2C_4 & 1 \\ C_3 & C_3C_4 & 0 \end{bmatrix} \begin{bmatrix} h \\ v_z^p \\ x \end{bmatrix} + \begin{bmatrix} C_1 \\ -C_2 \\ -C_3 \end{bmatrix} h_B + \begin{bmatrix} 0 \\ a_z^p + g + \omega_y^p v_x^p - \omega_x^p v_y^p \\ 0 \end{bmatrix} \quad (74)$$

Note that h is defined to be positive upward for convenience, v_z^p is positive downward, and x is the second derivative of h.

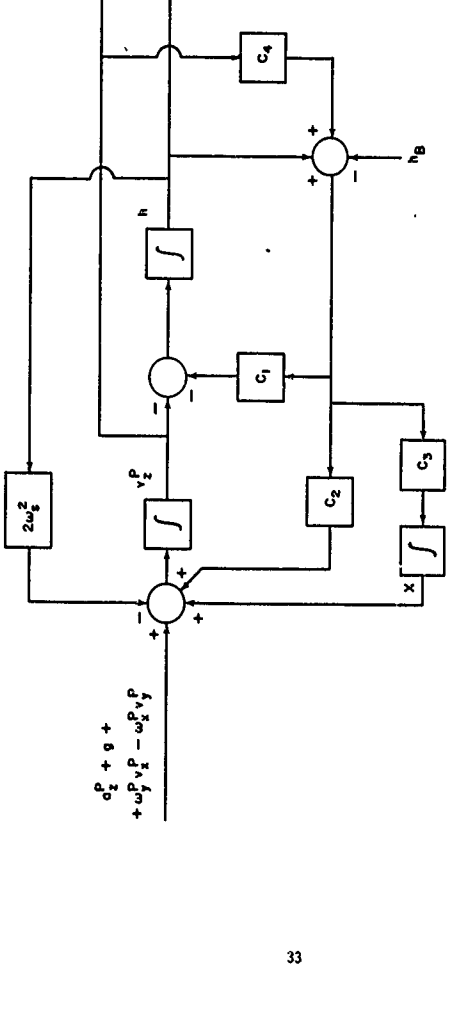


Figure 5. Third Order Damped Vertical Channel

The values of C_1, C_2, and C_3 are set so that the system characteristic equation yields three poles at $(-1/\tau)$,

$$(S + 1/\tau)^3 = 0 \tag{75}$$

where τ is the desired vertical channel time constant, usually chosen to be about 200 seconds. The gain values that achieve these system poles are:

$$C_1 = \frac{2\omega_s^2 C_4 \tau^3 + 3\tau^2 + 3C_4 \tau + C_4^2}{\left(1 - 2\omega_s^2 C_4^2\right)\tau^3} \tag{76a}$$

$$C_2 = \frac{2\omega_s^2 \left(\tau^3 + 3C_4 \tau^2\right) + 3\tau + C_4}{\left(1 - 2\omega_s^2 C_4^2\right)\tau^3} \tag{76b}$$

$$C_3 = \frac{1}{\tau^3} \tag{76c}$$

Once the appropriate values for τ and C_4 are chosen, these relations yield a time invariant linear system equation. A discrete-time updating equation can then be derived by means of the state transition matrix technique. This will be discussed more fully in Section IV.1.

The h_B value provided by the air data system can be calculated in the standard manner, or by means of the technique developed by R.L. Blanchard (Reference 1). Rather than calculating pressure a titude based on a standard atmosphere model (and possibly compensating later for non-standard conditions), this method computes a dynamically corrected altitude by numerically integrating the physical relation for a column of air as a function of pressure, gravity, and absolute temperature. The algorithm has already been flight tested and has demonstrated very accurate performance. Letting T_{ad} and P_{ad} be the absolute temperature

and pressure available from the air data system, the Blanchard algorithm with trapezoidal integration can be written as

$$\begin{aligned} D &= T_{ad}/P_{ad} \\ g &= g_0 + g_1 \sin^2 L + g_2 h_B \\ h_B &= h_B - C/g \left(D + D_{OLD}\right)\left(P_{ad} - P_{OLD}\right) \\ P_{OLD} &= P_{ad} \\ D_{OLD} &= D \end{aligned} \tag{77}$$

In the above equations, g_0, g_1, and g_2 are the values given in Equation 73, and C equals the universal gas constant R divided by twice the molecular weight of the atmosphere, 2M. Again it is noted that h_B, rather than h, is used to evaluate g.

7. INFORMATION EXTRACTION

The values of latitude L, longitude λ, and wander azimuth α are derived from the six propagated elements of $\underline{C}\,{}^{p}_{e}$ as in Equations 11, 12, and 13, using Equation 14 and the logic of Table I. Similarly, $(\psi + \alpha)$, θ, and ϕ are computed from the elements of $\underline{C}\,{}^{p}_{b}$ using Equations 18, 19, and 20, and the logic of Table II. Since these values are not required in the computational loop itself, the data extraction can be performed at any suitable iteration rate without any effect upon algorithm performance.

SECTION IV

ONBOARD IMPLEMENTATION

1. INTEGRATION TECHNIQUES

A number of propagation relations require integration in this system algorithm. Due to computer loading considerations, first order integration techniques will be used wherever they provide adequate performance. However, updating the elements of $\underline{C}_b^p$ will necessitate consideration of higher order techniques because of the potentially high angular rates involved. Previously the approach has been to employ DDA's operating at very high rates, as about 10,000 iterations per second, to provide accuracy with a first order technique. In contrast, this program uses a whole word computer operating at a slower iteration rate, but using higher order integration methods to achieve the desired precision.

There are various means of generating higher order techniques, such as the Runge-Kutta method and Taylor Series. Runge-Kutta techniques have been chosen because they provide an accuracy equivalent to that of a Taylor Series solution of the same order, while not requiring past histories of the dependent variables or high order derivative information.

Consider the updating of the direction cosine matrices $\underline{C}_b^p$ and $\underline{C}_e^p$. Either the four quaternion parameters for $\underline{C}_b^p$ or the six direction cosines for $\underline{C}_e^p$ (three are not propagated) can be arranged into a vector, denoted as $\underline{x}(t)$. Then an expression for the time rate of change of that vector, $\dot{\underline{x}}(t)$, can be written as

$$\dot{\underline{x}}(t) = \underline{f}\left[\underline{x}(t), \underline{\omega}(t)\right] \tag{78}$$

For the case of updating $\underline{C}_b^p$ using $\underline{\omega}_{pb}^b$, the quaternion relations corresponding to Equation 78 are

$$\dot{A} = \frac{1}{2}\left(\omega_z B - \omega_y C - \omega_x D\right) \tag{79a}$$

$$\dot{B} = \frac{1}{2}\left(-\omega_z A + \omega_x C - \omega_y D\right) \tag{79b}$$

$$\dot{C} = \frac{1}{2}\left(\omega_y A - \omega_x B - \omega_z D\right) \tag{79c}$$

$$\dot{D} = \frac{1}{2}\left(\omega_x A + \omega_y B + \omega_z C\right) \tag{79d}$$

where $\underline{x}^T = (A, B, C, D)$. Updating $\underline{C}_e^p$ with $\underline{\omega}_{ep}^p$ entails the component relations

$$\dot{C}_{12} = -\omega_y C_{32} \tag{80a}$$

$$\dot{C}_{13} = -\omega_y C_{33} \tag{80b}$$

$$\dot{C}_{22} = \omega_x C_{32} \tag{80c}$$

$$\dot{C}_{23} = \omega_x C_{33} \tag{80d}$$

$$\dot{C}_{32} = \omega_y C_{12} - \omega_x C_{22} \tag{80e}$$

$$\dot{C}_{33} = \omega_y C_{13} - \omega_x C_{23} \tag{80f}$$

where $\underline{x}^T = \left(C_{12}, C_{13}, C_{22}, C_{23}, C_{32}, C_{33}\right)$

Let the integration interval be T seconds long. Then a first order Runge-Kutta routine to solve for $\underline{x}(t + T)$ in terms of $\underline{x}(t)$ would be

$$\underline{x}(t + T) = \underline{x}(t) + T\ \underline{f}\left[\underline{x}(t), \underline{\omega}(t)\right] \tag{81}$$

This is the same as a first order Taylor Series solution or a first order approximation to a transition matrix solution to a set of linear equations.

A second order Runge-Kutta algorithm would be solved as

$$\underline{y} = \underline{x}(t) + T\ \underline{f}\left[\underline{x}(t), \underline{\omega}(t)\right] \tag{82a}$$

$$\underline{x}(t + T) = \underline{x}(t) + T/2\left\{\underline{f}\left[\underline{x}(t), \underline{\omega}(t)\right] + \underline{f}\left[\underline{y}, \underline{\omega}(t + T)\right]\right\} \tag{82b}$$

Finally, a fourth order Runge-Kutta technique computes $\underline{x}(t+T)$ according to the following algorithm:

$$\underline{\alpha} = T\ \underline{f}\left[\underline{x}(t), \underline{\omega}(t)\right] \tag{83a}$$

$$\underline{\beta} = T\ \underline{f}\left[\underline{x}(t) + \tfrac{1}{2}\underline{\alpha}, \underline{\omega}(t + T/2)\right] \tag{83b}$$

$$\underline{\gamma} = T\ \underline{f}\left[\underline{x}(t) + \tfrac{1}{2}\underline{\beta}, \underline{\omega}(t + T/2)\right] \tag{83c}$$

$$\underline{\delta} = T\ \underline{f}\left[\underline{x}(t) + \underline{\gamma}, \underline{\omega}(t + T)\right] \tag{83d}$$

$$\underline{x}(t + T) = \underline{x}(t) + \tfrac{1}{6}(\underline{\alpha} + 2\underline{\beta} + 2\underline{\gamma} + \underline{\delta}) \tag{83e}$$

The current algorithm formulation employs fourth order integration of the quaternion parameters related to $\underline{C}_b^p$ and first order integration of the elements of $\underline{C}_e^p$.

2. RATE EXTRACTION

All of the integration techniques require a knowledge of the angular rate $\underline{\omega}(t)$. In the case of updating $\underline{C}_b^p$, $\underline{\omega}_{pb}^b$ is generated in part by the output of the gyros, $\underline{\omega}_{ib}^b$, as seen in Equation 40. However, the gyro outputs are in the form of a pulse rate proportional to angular rate, and thus a pulse counter yields incremental change in angular orientation about the j-th axis, $\Delta\theta_j$. Let T be the iteration period of the integration routine. If $\Delta\theta_j(t, t + T)$ denotes the $\Delta\theta_j$ corresponding to the number of pulses counted between the times t and (t +T), a first order rate extraction would be computed as

$$\omega_j(t) = 1/T\ \Delta\theta_j(t, t + T) \tag{84}$$

This would be adequate for a first order integration method.

Since the fourth order algorithm (Equation 83) requires rate information at the midpoint of the integration interval, $\omega_j(t + T/2)$, a second order rate extraction is appropriate. The gyro output pulse

counter is sampled at twice the iteration frequency, every T/2 seconds. Thus the measurements $\Delta\theta_j(t, t + T/2)$ and $\Delta\theta_j(t + T/2, t + T)$ are used to calculate $\omega_j(t)$, $\omega_j(t + T/2)$, and $\omega_j(t + T)$ as required by Equation 83. Curve fitting of a second order polynomial to the data produces the following rate extraction relations:

$$\omega_j(t) = 1/T\left[3\Delta\theta_j(t, t + T/2) - \Delta\theta_j(t + T/2, t + T)\right] \quad (85a)$$

$$\omega_j(t + T/2) = 1/T\left[\Delta\theta_j(t, t + T/2) + \Delta\theta_j(t + T/2, t + T)\right] \quad (85b)$$

$$\omega_j(t + T) = 1/T\left[-\Delta\theta_j(t, t + T/2) + 3\Delta\theta_j(t + T/2, t + T)\right] \quad (85c)$$

The generation of the appropriate values of $\underline{\omega}_{ip}^{b}$ to be subtracted from the results of Equation 85 will be discussed in Section IV.4.

3. PROCESSING RATES

In order to maintain accuracy, the $\underline{C}_{b}^{p}$ matrix must be updated at a high frequency. However, the other computations need not be propagated at such a high iteration rate.

Consequently, the algorithm will be partitioned into segments that are iterated at different rates. Let T be the iteration period of the computation of $\underline{C}_{b}^{p}$. Then the gyro pulse counters are sampled every T/2 seconds, while the accelerometer pulse counter data is taken every T seconds. The other loop computations will be iterated every NT seconds, where N is an integer (for timing purposes in a digital computer, N might be set most conveniently as some power of two).

As has been mentioned previously, the information extraction (i.e., evaluations of $(\psi + \alpha)$, θ, ϕ, L, λ, and α) can occur at a completely different rate. For timing convenience, its period will be assumed equal to some integral number times T, denoted as KT. An efficient technique would be to let K = 2N, and to alternate outputting [L, λ, α] and [$(\psi + \alpha)$, θ, ϕ] on successive passes of the slower loop iterations.

4. FINAL ALGORITHM

Based upon the previous discussions, the overall system is partitioned as in Figure 6. The value at the top left of each segment is the period of that iteration cycle. Using this diagram, the interfaces and passage parameters between the various blocks will be described briefly.

First, the T/2 block provides $\underline{\Delta\theta}_{ib}^{b}(t, t + T/2)$ and $\underline{\Delta\theta}_{ib}^{b}(t + T/2, t + T)$ to the rate extraction algorithm for use in the integration interval $(t, t + T)$. These values are read into six memory registers reserved for gyro data, for subsequent use by the T block.

The other input to the T block is $\underline{\omega}_{ip}^{p}$, a slowly varying parameter generated every N iterations of the T block. Every T seconds, the direction cosine matrix $\underline{C}_{b}^{p}$ is recomputed, and a new value of $\underline{\omega}_{ip}^{b}$ is subsequently calculated based on the new $\underline{C}_{b}^{p}$ and the currently available value of $\underline{\omega}_{ip}^{p}$.

The output of the T block to the NT block is the sum of the most recent N values of $\underline{\Delta v}^{p}(t, t + T)$. Every N iterations by the T block, these sums are sent to the NT block to drive the velocity update equations.

Finally, the 2NT blocks are merely operations upon the current elements of the matrices $\underline{C}_{b}^{p}$ and $\underline{C}_{e}^{p}$.

The remainder of this section presents the detailed algorithm. All equations are presented explicitly in their scalar form.

a. The Partition Iterated Every T/2 Seconds

Let t be some integral number of periods T from initialization. The T/2 block provides the two samples of the gyro output pulse counters required every T seconds. When a counter is read, its contents are

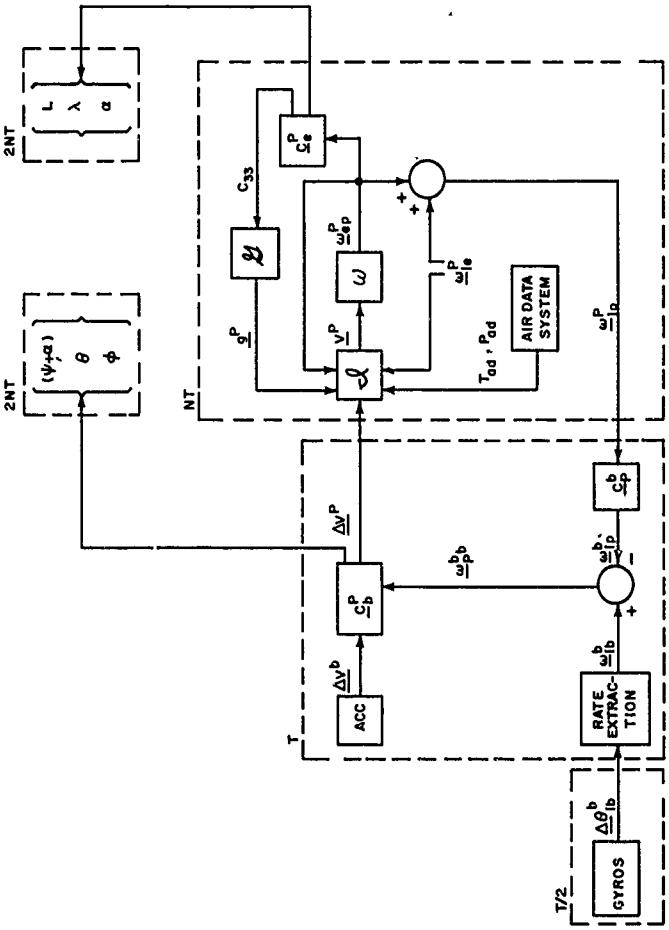


Figure 6. System Block Diagram

assumed to be set to zero. A flag is maintained (denoted here as IPASS but actually implemented as a countdown from the computer clock) to denote whether the current sample is of the form

$$\underline{\Delta 1} = \underline{\Delta\theta}_{ib}^{b}(t, t + T/2) \tag{86a}$$

or

$$\underline{\Delta 2} = \underline{\Delta\theta}_{ib}^{b}(t + T/2, t + T) \tag{86b}$$

If IPASS equals 2, the accelerometer outputs are read immediately after the gyro outputs, in order to minimize the time difference of the two samples. Thus, although the accelerometers are sampled every T seconds, the measurement appears in the T/2 portion of the algorithm. Figure 7 portrays the T/2 block calculations.

b. The Partition Iterated Every T Seconds

Now consider the T block. Figure 8 depicts the rate extraction algorithm that calculates $\underline{\omega}_{ib}^{b}$ at times t, (t + T/2), and (t + T) from the gyro data $\underline{\Delta 1}$ and $\underline{\Delta 2}$. At time (t + T), the quantities available for updating $\underline{C}_{b}^{p}$ are these three values of $\underline{\omega}_{ib}^{b}$ and a value of $\underline{\omega}_{ip}^{b}$ based on the current value of $\underline{\omega}_{ip}^{p}$ and the $\underline{C}_{b}^{pT}$ of the previous T block iteration. The single value of $\underline{\omega}_{ip}^{b}$ is subtracted from each of the three $\underline{\omega}_{ib}^{b}$ values to approximate the three corresponding values of $\underline{\omega}_{pb}^{b}$. Noting that

$$\underline{C}_{b}^{p} = \begin{bmatrix} T_{11} & T_{12} & T_{13} \\ T_{21} & T_{22} & T_{23} \\ T_{31} & T_{32} & T_{33} \end{bmatrix} \tag{87}$$

the matrix $\underline{C}_{p}^{b}$ can be written as

$$\underline{C}_{p}^{b} = \begin{bmatrix} T_{11} & T_{21} & T_{31} \\ T_{12} & T_{22} & T_{32} \\ T_{13} & T_{23} & T_{33} \end{bmatrix} \tag{88}$$

READ $\Delta\theta^{b}_{Ibx}$

READ $\Delta\theta^{b}_{Iby}$

READ $\Delta\theta^{b}_{Ibz}$

IF IPASS = 2, GO TO 2

$\Delta I_x = \Delta\theta^{b}_{Ibx}$

$\Delta I_y = \Delta\theta^{b}_{Iby}$

$\Delta I_z = \Delta\theta^{b}_{Ibz}$

IPASS = 2

RETURN TO EXECUTIVE

2 READ Δv^{b}_{x}

READ Δv^{b}_{y}

READ Δv^{b}_{z}

$\Delta 2_x = \Delta\theta^{b}_{Ibx}$

$\Delta 2_y = \Delta\theta^{b}_{Iby}$

$\Delta 2_z = \Delta\theta^{b}_{Ibz}$

IPASS = 1

GO TO T BLOCK

Figure 7. The T/2 Block

$$\omega^b_{ibx}(t) = 1/T\ (3\Delta 1_x - \Delta 2_x)$$

$$\omega^b_{iby}(t) = 1/T\ (3\Delta 1_y - \Delta 2_y)$$

$$\omega^b_{ibz}(t) = 1/T\ (3\Delta 1_z - \Delta 2_z)$$

$$\omega^b_{ibx}(t + T/2) = 1/T\ (\Delta 1_x + \Delta 2_x)$$

$$\omega^b_{iby}(t + T/2) = 1/T\ (\Delta 1_y + \Delta 2_y)$$

$$\omega^b_{ibz}(t + T/2) = 1/T\ (\Delta 1_z + \Delta 2_z)$$

$$\omega^b_{ibx}(t + T) = 1/T\ (3\Delta 2_x - \Delta 1_x)$$

$$\omega^b_{iby}(t + T) = 1/T\ (3\Delta 2_y - \Delta 1_y)$$

$$\omega^b_{ibz}(t + T) = 1/T\ (3\Delta 2_z - \Delta 1_z)$$

Figure 8. Rate Extraction

Using this convention, Figure 9 presents the computation of $\underline{\omega}^{b}_{ip}$ and the three values of $\underline{\omega}^{b}_{pb}$. In the $\underline{\omega}^{b}_{pb}$ relations, the following variables have been defined:

$$\underline{\omega 1} = \underline{\omega}^{b}_{pb}(t) \tag{89a}$$

$$\underline{\omega 2} = \underline{\omega}^{b}_{pb}(t + T/2) \tag{89b}$$

$$\underline{\omega 3} = \underline{\omega}^{b}_{pb}(t + T) \tag{89c}$$

Also, the rate extraction, $\underline{\omega}^{b}_{ip}$ and $\underline{\omega}^{b}_{pb}$ blocks are not combined because $\underline{\omega}^{b}_{ib}$ is used explicitly by alignment procedures, and thus should be separated to prevent redundant programming.

Figure 10 presents the $\underline{C}^{p}_{b}$ update, employing fourth order Runge-Kutta integration of four quaternion parameters. The initial value of $\underline{C}^{p}_{b}$ is available from the alignment routines, and its elements can be related to the initial Euler angles and wander azimuth by Equation 17. From the initial elements of $\underline{C}^{p}_{b}$ can be generated the initial values of A, B, C, and D through Equations 74 and 76. The update entails calculation of the small changes in the quaternion parameters, ΔA, ΔB, ΔC, and ΔD, in single precision, followed by a double precision update to the quaternion parameters themselves and the elements of $\underline{C}^{p}_{b}$. Double precision is required to add or subtract quantities of widely varying magnitude, or to calculate the small difference of large numbers, without excessive performance degradation due to truncation. The quaternion parameters are calculated and stored in double precision, whereas the T_{ij} values are computed with double precision operations but are stored in single precision. This is because the T_{ij} values are not used in subsequent updating, but are computed algebraically from the updated A, B, C, and D. In fact, performance analyses may indicate that the T_{ij} values can adequately be calculated in single precision.

CALCULATE $\underline{\omega}^b_{ip}$:

$$\omega^b_{ipx} = T_{11}\,\omega^p_{ipx} + T_{21}\,\omega^p_{ipy} + T_{31}\,\omega^p_{ipz}$$

$$\omega^b_{ipy} = T_{12}\,\omega^p_{ipx} + T_{22}\,\omega^p_{ipy} + T_{32}\,\omega^p_{ipz}$$

$$\omega^b_{ipz} = T_{13}\,\omega^p_{ipx} + T_{23}\,\omega^p_{ipy} + T_{33}\,\omega^p_{ipz}$$

CALCULATE $\underline{\omega}^b_{pb}$:

$$\omega 1_x = \omega^b_{ibx}(t) - \omega^b_{ipx}$$

$$\omega 1_y = \omega^b_{iby}(t) - \omega^b_{ipy}$$

$$\omega 1_z = \omega^b_{ibz}(t) - \omega^b_{ipz}$$

$$\omega 2_x = \omega^b_{ibx}(t + T/2) - \omega^b_{ipx}$$

$$\omega 2_y = \omega^b_{iby}(t + T/2) - \omega^b_{ipy}$$

$$\omega 2_z = \omega^b_{ibz}(t + T/2) - \omega^b_{ipz}$$

$$\omega 3_x = \omega^b_{ibx}(t + T) - \omega^b_{ipx}$$

$$\omega 3_y = \omega^b_{iby}(t + T) - \omega^b_{ipy}$$

$$\omega 3_z = \omega^b_{ibz}(t + T) - \omega^b_{ipz}$$

Figure 9. Calculation of $\underline{\omega}^b_{ip}$ and $\underline{\omega}^b_{pb}$

$$\alpha_1 = T/2\,(\omega I_z \cdot B - \omega I_y \cdot C - \omega I_x \cdot D)$$
$$\alpha_2 = T/2\,(\omega I_x \cdot C - \omega I_z \cdot A - \omega I_y \cdot D)$$
$$\alpha_3 = T/2\,(\omega I_y \cdot A - \omega I_x \cdot B - \omega I_z \cdot D)$$
$$\alpha_4 = T/2\,(\omega I_x \cdot A + \omega I_y \cdot B + \omega I_z \cdot C)$$
$$AI = A + \tfrac{1}{2}\alpha_1$$
$$BI = B + \tfrac{1}{2}\alpha_2$$
$$CI = C + \tfrac{1}{2}\alpha_3$$
$$DI = D + \tfrac{1}{2}\alpha_4$$
$$\beta_1 = T/2\,(\omega 2_z \cdot BI - \omega 2_y \cdot CI - \omega 2_x \cdot DI)$$
$$\beta_2 = T/2\,(\omega 2_x \cdot CI - \omega 2_z \cdot AI - \omega 2_y \cdot DI)$$
$$\beta_3 = T/2\,(\omega 2_y \cdot AI - \omega 2_x \cdot BI - \omega 2_z \cdot DI)$$
$$\beta_4 = T/2\,(\omega 2_x \cdot AI + \omega 2_y \cdot BI + \omega 2_z \cdot CI)$$
$$A2 = A + \tfrac{1}{2}\beta_1$$
$$B2 = B + \tfrac{1}{2}\beta_2$$
$$C2 = C + \tfrac{1}{2}\beta_3$$
$$D2 = D + \tfrac{1}{2}\beta_4$$
$$\gamma_1 = T/2\,(\omega 2_z \cdot B2 - \omega 2_y \cdot C2 - \omega 2_x \cdot D2)$$
$$\gamma_2 = T/2\,(\omega 2_x \cdot C2 - \omega 2_z \cdot A2 - \omega 2_y \cdot D2)$$
$$\gamma_3 = T/2\,(\omega 2_y \cdot A2 - \omega 2_x \cdot B2 - \omega 2_z \cdot D2)$$
$$\gamma_4 = T/2\,(\omega 2_x \cdot A2 + \omega 2_y \cdot B2 + \omega 2_z \cdot C2)$$
$$A3 = A + \gamma_1$$
$$B3 = B + \gamma_2$$

Figure 10. $\underline{C}_b^p$ Update

$$C3 = C + \gamma_3$$

$$D3 = D + \gamma_4$$

$$\delta_1 = T/2\ (\omega 3_z \cdot B3 - \omega 3_y \cdot C3 - \omega 3_x \cdot D3)$$

$$\delta_2 = T/2\ (\omega 3_x \cdot C3 - \omega 3_z \cdot A3 - \omega 3_y \cdot D3)$$

$$\delta_3 = T/2\ (\omega 3_y \cdot A3 - \omega 3_x \cdot B3 - \omega 3_z \cdot D3)$$

$$\delta_4 = T/2\ (\omega 3_x \cdot A3 + \omega 3_y \cdot B3 + \omega 3_z \cdot C3)$$

$$\Delta A = 1/6\left[\alpha_1 + 2(\beta_1 + \gamma_1) + \delta_1\right]$$

$$\Delta B = 1/6\left[\alpha_2 + 2(\beta_2 + \gamma_2) + \delta_2\right]$$

$$\Delta C = 1/6\left[\alpha_3 + 2(\beta_3 + \gamma_3) + \delta_3\right]$$

$$\Delta D = 1/6\left[\alpha_4 + 2(\beta_4 + \gamma_4) + \delta_4\right]$$

* DOUBLE PRECISION *

$$A \longleftarrow A + \Delta A$$

$$B \longleftarrow B + \Delta B$$

$$C \longleftarrow C + \Delta C$$

$$D \longleftarrow D + \Delta D$$

$$T_{11} = A^2 - B^2 - C^2 + D^2$$

$$T_{12} = 2(AB + CD)$$

$$T_{13} = 2(AC - BD)$$

$$T_{21} = 2(AB - CD)$$

$$T_{22} = -A^2 + B^2 - C^2 + D^2$$

$$T_{23} = 2(BC + AD)$$

$$T_{31} = 2(AC + BD)$$

$$T_{32} = 2(BC - AD)$$

$$T_{33} = -A^2 - B^2 + C^2 + D^2$$

Depicted in Figure 11 is the transformation of the accelerometer data into the wander azimuth coordinate frame. The sum registers are initialized at zero, and at the end of each T block loop, the components of $\underline{\Delta v}^p(t, t + T)$ are added to them; when these sums are used by the NT block, the register contents are reset to zero.

c. The Partition Iterated Every NT Seconds

Now the NT block calculations will be delineated. The unaided inertial system velocity updating equations were presented as Equations 55 and 56 of Section III.2. Further, the vertical channel damping was described by means of Equation 74, using Equation 76 to define numerical values involved, and Equation 77 to compute h_B. From Equation 74 can be derived a discrete-time update by using a first order Runge-Kutta (transition matrix approximation) method. This is modified slightly to express the result in terms of $\underline{\Delta v}^p(t, t + NT)$, as available from the sum registers of the T block. By defining

$$\begin{aligned}
\phi_{11} &= -NTC_1 & b_1 &= NTC_1 \\
\phi_{12} &= -(1 + C_1C_4)NT & b_2 &= -NTC_2 \\
\phi_{21} &= (C_2 - 2\omega_s^2)NT & b_3 &= -NTC_3 \\
\phi_{22} &= NTC_2C_4 \\
\phi_{23} &= NT \\
\phi_{31} &= NTC_3 \\
\phi_{32} &= NTC_3C_4
\end{aligned} \qquad (90)$$

where the C_i values are established in Section III.6 and ω_s^2 is the square of Schuler frequency, equal to $1.536217230 \times 10^{-6} rad^2/sec^2$, the $\mathcal{J}$ block algorithm can be developed as in Figure 12. First the air data system is sampled. Then the change in $\underline{v}^p$ is calculated in single precision and added to the previous value of $\underline{v}^p$ in double precision ("D.P." means double precision). Similar separation of single and double precision is maintained in the vertical loop calculation of altitude as well. Once the contents of the T block sum registers are used, these registers are zeroed for the next iteration. The values of g, $\underline{\omega}^p_{ep}$ and $\underline{\omega}^p_{ie}$ used in this algorithm are available from the previous NT block iteration.

$$\Delta v_x^p = T_{11}\,\Delta v_x^b + T_{12}\,\Delta v_y^b + T_{13}\,\Delta v_z^b$$

$$\Delta v_y^p = T_{21}\,\Delta v_x^b + T_{22}\,\Delta v_y^b + T_{23}\,\Delta v_z^b$$

$$\Delta v_z^p = T_{31}\,\Delta v_x^b + T_{32}\,\Delta v_y^b + T_{33}\,\Delta v_z^b$$

$$\Sigma_x^p \leftarrow \Sigma_x^p + \Delta v_x^p$$

$$\Sigma_y^p \leftarrow \Sigma_y^p + \Delta v_y^p$$

$$\Sigma_z^p \leftarrow \Sigma_z^p + \Delta v_z^p$$

Figure 11. Computation of $\underline{\Delta v}^p$

READ T_{ad}, P_{ad}

$$\omega_x^p = 2\omega_{iex}^p + \omega_{epx}^p$$

$$\omega_y^p = 2\omega_{iey}^p + \omega_{epy}^p$$

$$\omega_z^p = 2\omega_{iez}^p$$

$$\Delta V_x^p = \Sigma_x + NT\left(\omega_z^p V_y^p - \omega_y^p V_z^p\right)$$

$$\Delta V_y^p = \Sigma_y + NT\left(\omega_x^p V_z^p - \omega_z^p V_x^p\right)$$

$$\Delta V_z^p = \Sigma_z + NT\left(\omega_y^p V_x^p - \omega_x^p V_y^p + g\right)$$

$$\Sigma_x = 0$$

$$\Sigma_y = 0$$

$$\Sigma_z = 0$$

$$V_x^p \longleftarrow V_x^p + \Delta V_x^p \qquad \text{D.P.}$$

$$V_y^p \longleftarrow V_y^p + \Delta V_y^p \qquad \text{D.P.}$$

$$D = T_{ad}/P_{ad}$$

$$h_B \longleftarrow h_B - (C/g)(D + D_{OLD})(P_{ad} - P_{OLD}) \qquad \text{D.P.}$$

$$P_{OLD} = P_{ad}$$

$$D_{OLD} = D$$

$$y_1 = \phi_{11} h + \phi_{12} V_z^p + b_1 h_B$$

$$y_2 = \phi_{21} h + \phi_{22} V_z^p + \phi_{23} X + b_2 h_B + \Delta V_z^p$$

$$y_3 = \phi_{31} h + \phi_{32} V_z^p + b_3 h_B$$

$$h \longleftarrow h + y_1 \qquad \text{D.P.}$$

$$V_z^p \longleftarrow V_z^p + y_2 \qquad \text{D.P.}$$

$$X \longleftarrow X + y_3 \qquad \text{D.P.}$$

Figure 12. The $\mathcal{J}$ Block

The initial conditions for this integration are

$$
\begin{aligned}
h_B &= h_o && (+ \text{ UP}) \\
P_{OLD} &= P_o \\
D_{OLD} &= T_o/P_o \\
h &= h_{o\ INERTIAL} && (+ \text{ UP}) \\
V_z^p &= V_{zo\ INERTIAL}^p && (+ \text{ DOWN}) \\
X &= 0 \\
V_x^p &= V_{xo\ INERTIAL}^p \\
V_y^p &= V_{yo\ INERTIAL}^p
\end{aligned}
\qquad (91)
$$

where h_o is the altitude used to initialize the altimeter, and the subscript INERTIAL refers to the initial conditions in the inertial system. For ground alignment, the known altitude of the runway would be set into the air data and inertial systems, with velocities and accelerations zero.

If h_B is to be read directly from the air data system, the first four statements after $v_y^p \leftarrow v_y^p + \Delta v_y^p$ can be removed. By replacing the statements after $v_y^p \leftarrow v_y^p + \Delta v_y^p$ with the relation $v_z^p \leftarrow v_z^p + \Delta v_z^p$, an unaided inertial system is obtained.

The ω segment, depicted in Figure 13, computes $\underline{\omega}_{ep}^{p}$ from $\underline{v}^p$ using the $\underline{C}_e^p$ elements generated in the previous NT block iteration. In these relations the constants are the reciprocal of earth equatorial radius and ellipticity:

$$\frac{1}{r_o} = \frac{1}{20{,}926{,}488 \text{ ft.}} = 4.778632707 \times 10^{-8} \text{ ft}^{-1} \qquad (92)$$

$$e = \frac{1}{297.00} = 3.367003367 \times 10^{-3} \qquad (93)$$

ω BLOCK

$$R_y^{-1} = \frac{1}{r_0}\left[1 - \frac{1}{r_0}h - e\left(1 - 3C_{23}^2 - C_{13}^2\right)\right]$$

$$R_x^{-1} = \frac{1}{r_0}\left[1 - \frac{1}{r_0}h - e\left(1 - 3C_{13}^2 - C_{23}^2\right)\right]$$

$$k = \frac{2e}{r_0} C_{13} C_{23}$$

$$\omega_{epx}^p = R_y^{-1} V_y^p + k V_x^p$$

$$\omega_{epy}^p = -R_x^{-1} V_x^p - k V_y^p$$

$\underline{C}_e^p$ BLOCK

$$\Delta C_{12} = -NT\left(\omega_{epy}^p C_{32}\right)$$

$$\Delta C_{13} = -NT\left(\omega_{epy}^p C_{33}\right)$$

$$\Delta C_{22} = NT\left(\omega_{epx}^p C_{32}\right)$$

$$\Delta C_{23} = NT\left(\omega_{epx}^p C_{33}\right)$$

$$\Delta C_{32} = NT\left(\omega_{epy}^p C_{12} - \omega_{epx}^p C_{22}\right)$$

$$\Delta C_{33} = NT\left(\omega_{epy}^p C_{13} - \omega_{epx}^p C_{23}\right)$$

* DOUBLE PRECISION *

$$C_{12} \longleftarrow C_{12} + \Delta C_{12}$$

$$C_{13} \longleftarrow C_{13} + \Delta C_{13}$$

$$C_{22} \longleftarrow C_{22} + \Delta C_{22}$$

$$C_{23} \longleftarrow C_{23} + \Delta C_{23}$$

$$C_{32} \longleftarrow C_{52} + \Delta C_{32}$$

$$C_{33} \longleftarrow C_{33} + \Delta C_{33}$$

Figure 13. ω and $\underline{C}_e^p$ Blocks

Figure 13 also depicts the first order integration of $\underline{C}^{p}_{e}$, in which the changes to the six direction cosines are computed in single precision and then added to the whole values in double precision. The C_{ij} values thus are stored in double precision, with the most significant half used in subsequent single precision computations.

The remainder of the NT block calculations are shown in Figure 14. The earth rate vector $\underline{\omega}^{p}_{ie}$ is generated with the newly updated elements of $\underline{C}^{p}_{e}$ and the constant ω_{ie}, 7.2921160×10^{-5} rad/sec. Having generated $\underline{\omega}^{p}_{ep}$ and $\underline{\omega}^{p}_{ie}$, $\underline{\omega}^{p}_{ip}$ is then formed as their sum, to be used as an output to the T block. Finally, g is calculated with the current value of C_{33} and the value of h_B from the $\mathcal{G}$ block.

d. Data Extraction

As presently programmed, data is extracted from $\underline{C}^{p}_{e}$ on odd numbered iterations of the NT block, and from $\underline{C}^{p}_{b}$ on even numbered iterations. THe $\underline{C}^{p}_{e}$ extraction is shown in Figure 15, and that from $\underline{C}^{p}_{b}$ in Figure 16. As is noted in these figures, double precision may be required in evaluating the $\sin^{-1}$ and $\tan^{-1}$ functions to provide adequate resolution.

CALCULATE $\underline{\omega}^{p}_{ie}$:

$$\omega^{p}_{iex} = \omega_{ie} C_{13}$$
$$\omega^{p}_{iey} = \omega_{ie} C_{23}$$
$$\omega^{p}_{iez} = \omega_{ie} C_{33}$$

CALCULATE $\underline{\omega}^{p}_{ip}$:

$$\omega^{p}_{ipx} = \omega^{p}_{iex} + \omega^{p}_{epx}$$
$$\omega^{p}_{ipy} = \omega^{p}_{iey} + \omega^{p}_{epy}$$
$$\omega^{p}_{ipy} = \omega^{p}_{iez}$$

CALCULATE g:

$$g = 32.087437360 + 0.16994493815\, C_{33}^{2} - 0.30877321095 \times 10^{-5} h_B$$

Figure 14. Calculation of $\underline{\omega}^{p}_{ie}$, $\underline{\omega}^{p}_{ip}$ and $\underline{g}$

$L = \sin^{-1}(-C_{33})$

$C_{31} = C_{23}C_{12} - C_{22}C_{13}$

$\lambda = \tan^{-1}(C_{32}/C_{31})$

IF (C_{31} = + and λ = +), $\lambda \leftarrow \lambda - 180°$

IF (C_{31} = + and λ = −), $\lambda \leftarrow \lambda + 180°$

$\alpha = \tan^{-1}(C_{23}/C_{13})$

IF (C_{13} = −), $\alpha \leftarrow \alpha + 180$

IF (C_{13} = + and α = −), $\alpha \leftarrow \alpha + 360$

$\sin^{-1}(\cdot)$ and $\tan^{-1}(\ )$ MAY REQUIRE DOUBLE PRECISION

Figure 15. 2Nt Block Associated with $\underline{C}_e^p$

$\theta = \sin^{-1}(-T_{31})$

$\phi = \tan^{-1}(T_{32}/T_{33})$

IF ($T_{33} = -$ and $\phi = +$), $\phi \leftarrow \phi - 180°$

IF ($T_{33} = -$ and $\phi = -$), $\phi \leftarrow \phi + 180°$

$\psi_T = \tan^{-1}(T_{21}/T_{11})$

IF ($T_{11} = -$), $\psi_T \leftarrow \psi_T + 180°$

IF ($T_{11} = +$ and $\psi_T = -$), $\psi_T \leftarrow \psi_T + 360°$

$\psi = \psi_T - \alpha$

IF ($\psi = -$), $\psi \leftarrow \psi + 360°$

$\sin^{-1}(\cdot)$ and $\tan^{-1}(\cdot)$ MAY REQUIRE
DOUBLE PRECISION

Figure 16. 2NT Block Associated with $\underline{C}_b^p$

SECTION V

USE OF ALGORITHM

This report has developed the algorithm to derive attitude and position information from the outputs of strapdown gyros and accelerometers. By choosing appropriate values for the fast loop period T, the multiplicative factor N to obtain the slow rate, and the integration order to be employed for the various updates, a specified computational accuracy can be attained for any expected vehicle environment. Performance analyses can determine these parameter values for each application.

The specific program for which this algorithm was developed envisions an angular rate environment in excess of one radian per second, and so a fourth order integration of $\underline{C}_b^p$ was chosen. With first order integrations used for the other propagations, the values of T = 0.025 sec and N = 4 have been projected as yielding adequate performance. However, with other choices of these parameters, and specification of desired vertical channel time constant τ and altimeter lag C_4, the algorithm can be adapted to any strapdown application.

To achieve desirable system reliability, redundant strapdown sensors will be used in many prospective systems. The basic algorithm remains unchanged for this application. Logic networks would be included in the system to perform failure detection, faulty sensor isolation, and sensor recalibration and/or system reconfiguration. Thus, the sensor outputs are processed to attain a best representation of the quantities $\underline{\Delta\theta}_{ib}^b$ and $\underline{\Delta v}^b$ coordinatized along the three body axes, and these are then used as inputs to the algorithm.

Thus, the algorithm is flexible enough to be compatible with a variety of different vehicle, mission, and strapdown sensor package configurations. With a moderate increase in computer loading, the advantages that a strapdown system bears over a gimballed platform can be fully exploited in future aerospace vehicles.

REFERENCES

1. R. L. Blanchard, "A New Algorithm for Computing Inertial Altitude and Vertical Velocity," IEEE Transactions on Aerospace and Electronic Systems, Vol AES-7, No. 6, November 1971.

2. K. R. Britting, Inertial Navigation Systems Analysis, Wiley-Interscience, New York, 1971.

3. A. J. Brockstein and J. T. Kouba, "Derivation of Free Inertial, General Wander Azimuth Mechanization Equations," Publication No. 9176A, Litton Systems, Inc., Guidance and Control Systems Division, Woodland Hills, California, June 1969.

4. C. Broxmeyer, Inertial Navigation Systems, McGraw-Hill Book Co., New York, 1964.

5. H. Goldstein, Classical Mechanics, Addison-Wesley Publishing Co., Inc., Reading, Massachusetts, 1959.

6. F. J. Marcus, "Computational Comparison of Strapdown System Attitude Algorithms", Report TE-43, Measurement Systems Laboratory, Massachusetts Institute of Technology, Cambridge, Massachusetts, June 1971.

7. R. A. McKern, "A Study of Transformation Algorithms for Use in a Digital Computer," Report T-493, Massachusetts Institute of Technology Instrumentation Laboratory (now the C. S. Draper Laboratory), Cambridge, Massachusetts, January 1968.

8. J. C. Pinson, "Inertial Guidance for Cruise Vehicles", in Leondes, C. T., ed., Guidance and Control of Aerospace Vehicles, McGraw-Hill Book Co., New York, 1963.

9. T. F. Wiener, "Theoretical Analysis of Gimballess Inertial Reference Equipment Using Delta-Modulated Instruments", Report T-300, Massachusetts Institute of Technology Instrumentation Laboratory, Cambridge, Massachusetts, March 1962.

10. J. C. Wilcox, "A New Algorithm for Strapped-Down Inertial Navigation", IEEE Transactions on Aerospace and Electronics Systems, Vol AES-3, No. 5, September 1967.

11. W. Wrigley, W. M. Hollister, and W. G. Denhard, Gyroscopic Theory, Design, and Instrumentation, MIT Press, Cambridge, Massachusetts, 1969.